Mesoscopic simulations at the physics-chemistry-biology interface

Massimo Bernaschi

IAC-CNR, via dei Taurini 19, 00185, Rome, Italy

Simone Melchionna

ISC-CNR, c/o Dipartimento di Fisica, Università La Sapienza,
P. le A. Moro 5, 00185, Rome, Italy
and IACS-SEAS, Harvard University,
Oxford Street 29, Cambridge, Massachusetts 02138, USA

Sauro Succi

Center for Life Nano Science@Sapienza, Italian Institute of Technology,
Viale Regina Elena, 295, I-00161 Rome, Italy,
IAC-CNR, via dei Taurini 19, 00185, Rome, Italy,
and IACS-SEAS, Harvard University,
Oxford Street 29, Cambridge, Massachusetts 02138, USA

(published 28 May 2019)

This review discusses the lattice Boltzmann–particle dynamics (LBPD) multiscale paradigm for the simulation of complex states of flowing matter at the interface between physics, chemistry, and biology. In particular, current large-scale LBPD simulations of biopolymer translocation across cellular membranes, molecular transport in ion channels, and amyloid aggregation in cells are described. Prospects are provided for future LBPD explorations in the direction of cellular organization, the direct simulation of full biological organelles, all the way up to physiological scales of potential relevance to future precision-medicine applications, such as the accurate description of homeostatic processes. It is argued that, with the advent of Exascale computing, the mesoscale physics approach advocated in this review may come to age in the next decade and open up new exciting perspectives for physics-based computational medicine.

CONTENTS

I. INTRODUCTION

Thanks to the spectacular advances of computer technology (hardware and software), on the one hand, and mathematical modeling, on the other hand, in the last few decades modern science has come to the point of providing a quantitative description of many biological systems, whose complexity would have been regarded as impossible until only recently. In this review, we illustrate the point through several concrete examples.

Notwithstanding such major advances, the challenge of modeling biological and physiological systems remains formidable, as it actually amounts to cover some ten decades in space (from molecules to the human body) and easily twice as many in time. No mathematical or computational model in the foreseeable future can take up such a challenge head-on, i.e., by direct simulation of all the actual mechanisms, scales, and levels involved in the process.

Coarse-grained methods come in many flavors and families, depending on the range of scales and problems they are targeted to, but in this review we focus on a specific mesoscale technique, known as the lattice Boltzmann method (LBM), namely, a minimal lattice version of the Boltzmann equation (Cercignani and Berman, 1976; Boltzmann, 2012) which has witnessed a burgeoning growth for the description of complex flow phenomena across an impressively broad range of scales.

The lattice Boltzmann equation (LBE) was developed as a computational alternative to the discretization of the Navier-Stokes (NS) equations of continuum fluid mechanics (Benzi, Succi, and Vergassola, 1992; Succi, 2001).

Over the years, however, it has been proof of an amazing and largely unanticipated versatility and ability to describe a broad variety of phenomena involving complex states of flowing matter, beyond the strict realm of continuum hydrodynamics, including nontrivial flows at microscales and nanoscales. Thanks to this versatility, and to the coupling with various families of mesoscale particle methods, in the last decade the LBE has gained increased status for the simulation of many complex flow problems at the interface between fluid dynamics, chemistry, material science, and biology. These include, for instance, multiphase and multicomponent flows with complex interfaces or the motion of suspended bodies under strong geometrical confinement, possibly with chemical reactions (Krüger et al., 2017; Succi, 2018). After revisiting

the main ideas behind the lattice Boltzmann (LB) theory, in this review we discuss current and future prospects of multi-scale or level lattice Boltzmann–particle dynamics (LBPD) simulations at the physics-chemistry-biology interface in an attempt to identify and portray outstanding problems of potential relevance to clinical applications in the not-so-distant future, i.e., computational medicine.

Computer-assisted medicine is a consolidated branch of modern science, which generally develops around two separate pillars: molecular biology and macroscale physiology. The former is heavily leaning on bioinformatics and data science tools, while the latter usually relies upon the methods of continuum and fluid mechanics. In this review, we portray a third, alternative approach, based on mesoscale physics, i.e., physics-informed coarse-grained models of microscale biological processes, possibly embedded within their physiological environment. This mesoscale approach is grounded into the intermediate level of the description of matter, namely, kinetic theory, in both its versions, Boltzmann's kinetic theory and Langevin stochastic particle dynamics. As a result, the main tools of the mesoscale approach are fluids, particles, and probability distribution functions. The multiscale LBPD concept is thus apparent; Boltzmann's theory naturally connects upward to continuum hydrodynamics, while Langevin dynamics connects downward to the molecular level. Combining the two in a single computational harness opens up a direct route from the continuum to the atomistic word and back.

Several specific examples have already shown the potential of LBPD simulations in areas straddling across physics and biology. A selected set of applications is described in detail in this review to provide a taste for the breadth of applications that have been tackled in the recent past. Readers keen on specific details are directed to the original literature.

Clearly, the physical-chemical-biological (PCB) interface is enormously rich and varied, ranging from nanometric macromolecular phenomena, to peptidic aggregation and biopolymer translocation, to cellular motion and active matter, as sketched in Fig. 1. In a few words, the topic goes way beyond the scope of the present work. Nevertheless, it is
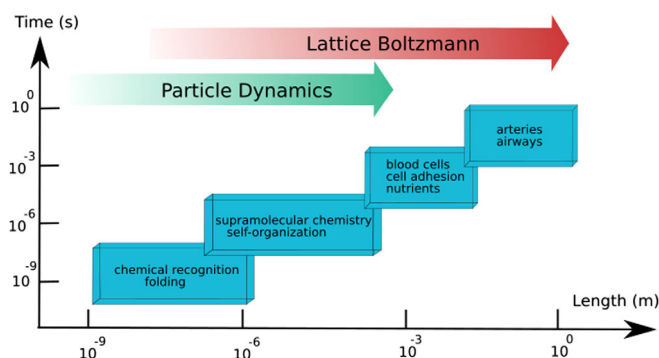
FIG. 1. Scales related to biological and medical applications, whose stretch can be covered by the LBPD approach. While the particle description enables the representation of complex macromolecules or cellular organizations by tracking the fate of each individual particle, the LB description is based on the collective motion of solvent molecules. The boundary between LB and PD is blurred and depends on the degree of microscopic detail required by each single application (see also Fig. 8).
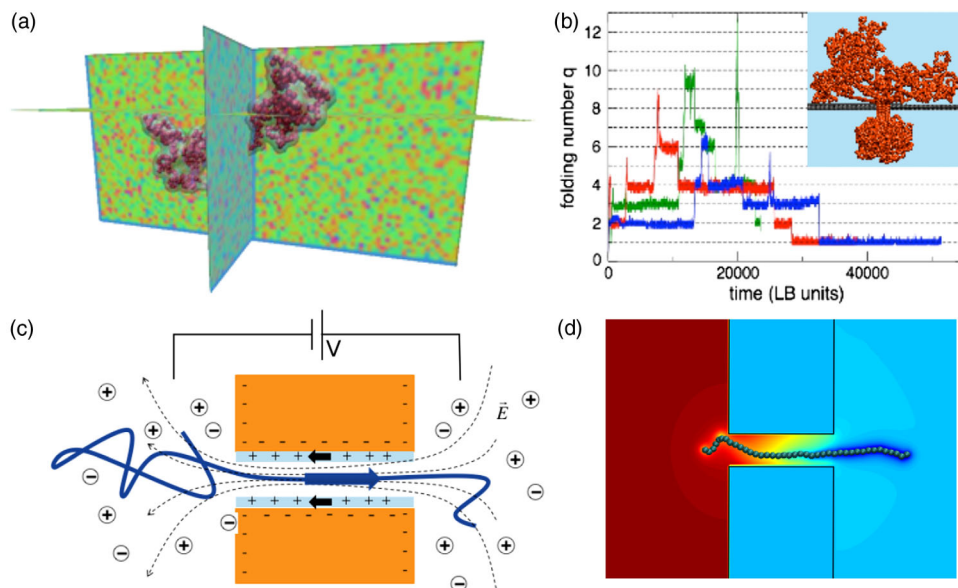
FIG. 2. Translocation of a biopolymer, as in the case of DNA, through a narrow pore. Different representations can be used to study multiple levels of detail during the translocation process. (a), (b) The biopolymer is represented as a simple necklace of beads by neglecting correlations stemming from the local molecular rigidity or backbone charge. (c), (d) At the next level, the macromolecule is charged and moves in an electrolyte solution, whereby a neutral solvent and counterions and coions migrate due to an externally applied electric field, giving rise to an electro-osmotic flow that ultimately causes the molecule to translocate. From Datar *et al.*, 2017.

important for one to appreciate the great flexibility of the method to cope with multiple scales of motion and, more importantly, its ability to incorporate the desired degree of biochemical specificity inherent to the problem at hand.

The LBPD framework can be enriched in the direction of describing complex flowing states of matter at microscales and nanoscales. Although not "rigorous," such variants prove capable of providing new physical insight into highly complex states of flowing matter. Remarkably, such extensions can be put in place without compromising the outstanding parallel amenability to parallel processing of the method. On the assumption that computing power will keep growing in the next few decades, if only perhaps at sub-Moore's paces, this state of affairs spawns tremendous opportunities to gain new insight into a series of fundamental problems dealing with complex states of flowing matter in general, and with a special focus on those relevant to biology and medicine.

This review is organized into three main parts. In the first one, we discuss the basic aspects of Boltzmann's kinetic theory with a special emphasis on its lattice version for fluid dynamics and its extensions to soft matter and biological applications, including the coupling to particle dynamics for the motion of suspended bodies.

In the second part, we describe selected applications to biological and physiological systems, such as biopolymer translocation, ion channels, protein diffusion, and amyloid aggregation in cellular environments. For a quick visual appreciation, see Figs. 2, 3, and 4 to be commented on in detail later.

In the third part, we provide a prospective view of a series of problems at the physics-chemistry-biology interface, which may become accessible once Exascale computers are with us. Special attention is paid to their potential import for clinical applications, such as the direct simulation of biological organelles and the quantitative description of hemostatic processes.

Finally, due to the crucial role played by high-performance computing in this story, in the Appendix we provide an extended account of the main issues involved with the implementation of the LBPD scheme on high-end parallel computers in the Exascale range.

The main message we want to convey in this review is that a *mesoscale physics-based approach to computational medicine* may come of age in the next decade.
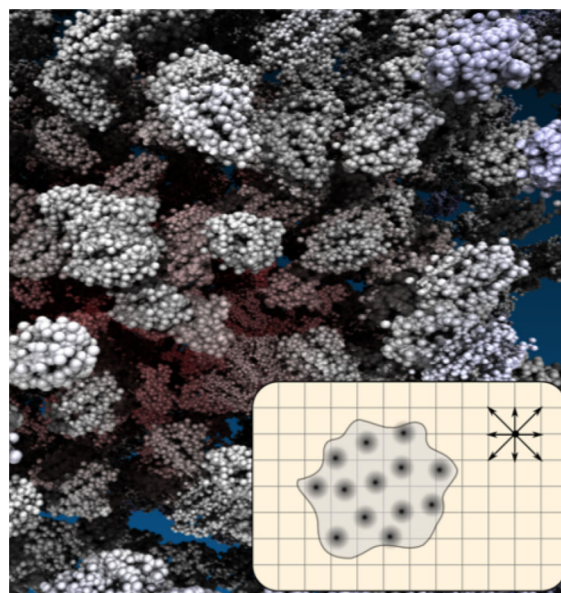

FIG. 3. Representation of the crowded interior of the cell as obtained from simulations. The inset illustrates the embedding of a protein in the LB mesh and each protein atom is represented via the DPM particle-fluid exchange scheme. From Bernaschi *et al.*, 2013a.
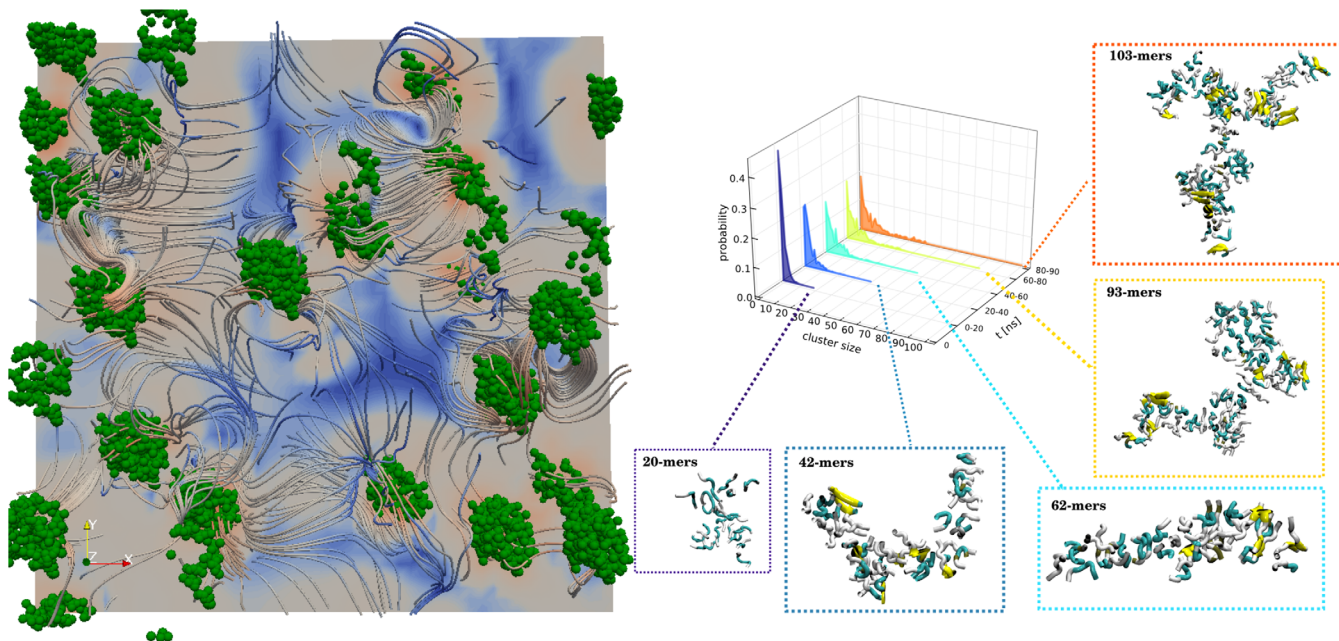
FIG. 4.   Left: Snapshot of a configuration of 1000 amyloid peptides $A\beta_{16-22}$ simulated in a cubic box of size $30 \times 30 \times 30$ nm^3 and complex flow structure generated by their motion in the surrounding solvent. Right: Evolution of the size of the peptidic aggregates as a function of time. From Nasica-Labouze *et al.*, 2015, and Chiricotto *et al.*, 2017.

II. BOLTZMANN KINETIC THEORY

The Boltzmann equation (BE) is the core of Boltzmann's kinetic theory that, in turn, is the cornerstone of nonequilibrium statistical mechanics, a pillar of theoretical physics at large (Boltzmann, 2012). Besides its paramount conceptual value as a bridge between the microscopic world of atoms and molecules and the macroscopic world of thermohydrodynamic fields, the BE also provides a concrete tool for the quantitative investigation of a broad variety of practical nonequilibrium transport problems (Cercignani and Berman, 1976). However, the BE is all but an easy piece to work with a nonlinear integrodifferential equation in $6 + 1$ (phase-space plus time) dimensions. This motivates a relentless search for new methods to solve the BE either analytically or numerically, the latter option usually covering a broader ground. Graeme Bird's direct simulation Monte Carlo (DSMC) method has played a leading role in this respect and continues to do so to this day (Bird, 1994). In principle, DSMC solves the BE directly and in full, i.e., accounting for the specificity of molecular interactions, as well as strong nonequilibrium effects, using a stochastic particle technique, whence the Monte Carlo label. This comes at a major computational cost, which is why various approximations have been developed and considerably refined over the years (Ketsdever and Struchtrup, 2016; Dimarco *et al.*, 2018). Close to local equilibrium and away from confining elements, however, molecular details become increasingly irrelevant. Universality takes stage and more economical descriptions can be devised. The basic idea is to relinquish the "irrelevant" details while still preserving the basic properties of macroscopic physics, namely, the symmetries and conservation laws which secure the emergence of the NS equations from the underlying molecular dynamics. Among others, a description

which has gained major interest for the last three decades is the LB method (Benzi, Succi, and Vergassola, 1992; Succi, 2001, 2018; Krüger *et al.*, 2017). The LB method was devised with the specific intent of providing an alternative to the discretization of the NS equations for the numerical solution of continuum hydrodynamic problems. This still is its mainstay and, for some, also the only place where it belongs. Indeed, the use of LB for flows beyond NS was ruled out (Luo, 2004; Junk, Klar, and Luo, 2005), mostly on account of the lack of a rigorous asymptotic limit. The above no go has been proven largely over-restrictive and current applications beyond the strict realm of continuum fluid dynamics abound, especially in the direction of soft matter. Since problems in biology and medicine hardly involve fluid mechanics alone, such developments are of direct relevance to computational explorations at the interface between physics, chemistry, and biology, the main scope of this review.

A. The Boltzmann equation

The BE of classic kinetic theory is basically a continuity equation in six-dimensional *phase space*, namely (Boltzmann, 2012),

$$\partial_t f + \vec{v} \cdot \nabla_r f + \vec{a} \cdot \nabla_v f = C(f, f), \qquad (1)$$

where $f \equiv f(\vec{r}, \vec{v}; t)$ is the probability density of finding a molecule at position $\vec{r}$ in ordinary space, with velocity $\vec{v}$ at time t. The left-hand side of the BE represents the streaming of the molecules under the effect of a force field $\vec{F} = m\vec{a}$ and reflects the Newtonian mechanics $d\vec{r}/dt = \vec{v}$ and $d\vec{v}/dt = \vec{F}/m$ of a representative molecule, while the right-hand side describes intermolecular interactions. For the case of a dilute

gas, as originally considered by Boltzmann, these interactions are typically taken in the form of two-body local collisions, since higher-order encounters are much less frequent, and hence negligible in the collision count. Even under such major simplifications, solving the BE presents a daunting challenge on account of its high-dimensionality, six phase-space dimensions plus time, as well as due to the nonlinear (quadratic) integral character of the collision operator (Cercignani and Berman, 1976).

Regardless of the complexity of the underlying microscopic interactions, the collision operator must comply with the mass-momentum-energy conservation laws, namely,

$$\int C(f,f)\{1, \vec{v}, v^2\}d\vec{v} = 0. \qquad (2)$$

In addition, it must also secure compliance with the second principle, which amounts to supporting a so-called H theorem, namely,

$$-\frac{d}{dt}\int f \log f d\vec{r}d\vec{v} \geq 0. \qquad (3)$$

In other words, the dynamics of the distribution function must converge to a universal global attractor, corresponding to the thermodynamic equilibrium.

The macroscopic fluid variables are obtained by a linear and local contraction of the Boltzmann distribution, namely,

$$\rho(\vec{r}, t) = \int f d\vec{v}, \qquad (4)$$

$$\rho\vec{u}(\vec{r}, t) = \int f\vec{v}d\vec{v}, \qquad (5)$$

$$\rho k_B T(\vec{r}, t) = \int fm(v-u)^2 d\vec{v} = 0, \qquad (6)$$

where ρ is the mass density, $\vec{u}$ the flow speed, and T the flow temperature in D spatial dimensions.

Central to the emergence of hydrodynamic behavior is the notion of *local equilibrium*. This is defined as the specific form attained by the Boltzmann distribution once collisions come in complete balance, i.e.,

$$C(f^{\text{eq}}, f^{\text{eq}}) = 0. \qquad (7)$$

Inspection of the Boltzmann collision operator provides the following universal Maxwell-Boltzmann (MB) local equilibrium distribution:

$$f^{\text{eq}}(\vec{v}, \vec{r}, t) = Z^{-1}\rho e^{-c^2/2}, \qquad (8)$$

where $Z = (2\pi v_T^2)^{D/2}$ is a normalization constant in D spatial dimensions, $v_T = \sqrt{k_B T/m}$ is the thermal speed, and

$$\vec{c} = \frac{\vec{v} - \vec{u}}{v_T}$$

is the peculiar speed, i.e., the molecular velocity relative to the fluid one, in units of the thermal speed. One familiar with statistical mechanics will readily recognize the canonical distribution $e^{-E/k_B T}$ in the comoving frame of the fluid, with the identification $E = mc^2/2$.

A few comments are in order. First, the MB distribution depends on space and time only through the hydrodynamic fields $\{n(\vec{r}, t), \vec{u}(\vec{r}, t), T(\vec{r}, t)\}$, its dependence on the velocity variable being a universal Gaussian distribution. This is a strict consequence of Eq. (2), i.e., the microscopic conservation laws.

Such dependence is largely arbitrary, with the caveat that it should be weak on the molecular scale. More precisely, the macrofields should not show appreciable changes on the scale of the molecular *mean-free path*, that is,

$$\text{Kn} \equiv \lambda \left| \frac{\nabla_r M}{M} \right| \ll 1, \qquad (9)$$

where M designates any macrofield and λ is the molecular mean-free path. The ratio (9), known as the Knudsen number, serves as the smallness parameter controlling the emergence of the hydrodynamic limit from Boltzmann's kinetic equation. Ordinary fluids dynamics holds in the range $\text{Kn} \sim 0.01$ and below.

B. From Boltzmann to Navier-Stokes hydrodynamics

The conceptual path from the BE to the NS equations of continuum fluids is based on two fundamental steps:
(1) Projection of the Boltzmann equation upon a suitable basis function in velocity space, typically Hermite polynomials in Cartesian coordinates.
(2) Multiscale expansion using the Knudsen number as a smallness parameter on the assumption of weak departure from local equilibrum.

The projection generates a hierarchy of partial differential equations for the kinetic moments:

$$\partial_t M_0 + \nabla \cdot M_1 = 0, \qquad (10)$$

$$\partial_t M_1 + \nabla \cdot M_2 = 0, \qquad (11)$$

$$\partial_t M_2 + \nabla \cdot M_3 = \frac{M_2^{\text{eq}} - M_2}{\tau}, \qquad (12)$$

where

$$M_n \equiv M_n(\vec{r}; t) = \int f(\vec{r}, \vec{v}; t)H_n(\vec{v})d\vec{v}, \qquad (13)$$

and $H_n(v)$ denotes the nth order tensor Hermite polynomial. Note that M_n is a tensor of rank n, namely, M_0 (scalar) is the fluid density, M_1 (vector) is the fluid current, and M_2 (second-order tensor) is the momentum flux, whose trace delivers twice the kinetic energy of the fluid and the triple tensor M_3 is the flux of momentum flux.

The left-hand side shows that the moment hierarchy is open, since the time derivative of M_n is driven by the divergence of M_{n+1}. This is the mechanism by which heterogeneities fuel

nonequilibrium. Also note that the right-hand side of the first two equations is zero because collisions conserve mass and momentum. However, they do not conserve momentum flux, which is why the right-hand side of the third equation is nonzero, expressing the relaxation of the momentum flux to its equilibrium expression. Such relaxation takes place on a collisional time scale τ, which in turn fixes the *kinematic viscosity* of the fluid. Instantaneous relaxation ($\tau \to 0$) denotes the infinitely strong collisional regime whereby collisions do not leave any chance for nonequilibrium to survive, formally corresponding to the idealized case of a perfect (zero dissipation) fluid. Macroscopically, this corresponds to the inviscid Euler equations.

All real fluids, though, relax in a short but finite time (strictly speaking this is also true for superfluids), and consequently the moment equations present an open hierarchy which needs to be closed somehow. This is where the assumption of weak departure from local equilibrium takes stage.

By that assumption, one formally expands the distribution function and space-time derivatives in powers of the Knudsen number, replaces the expansion in the moment equations, and collects homologue terms order by order in the Knudsen number. To zero order, the Euler equations are obtained, whereas the first order delivers the NS equations of dissipative fluids, namely,

$$\partial_t \rho + \nabla \cdot (\rho \vec{u}) = 0, \tag{14}$$

$$\partial_t (\rho \vec{u}) + \nabla \cdot (\rho \vec{u} \, \vec{u}) = -\nabla p + \nabla \cdot \overset{\leftrightarrow}{\sigma}, \tag{15}$$

where $\overset{\leftrightarrow}{\sigma}$ is a second-order tensor formed by spatial derivatives of the flow field. Typically,

$$\overset{\leftrightarrow}{\sigma} = 2\mu \overset{\leftrightarrow}{S} + \lambda d \overset{\leftrightarrow}{I}, \tag{16}$$

where $\overset{\leftrightarrow}{S} = (1/2)[\nabla \vec{u} + (\nabla \vec{u})^T]$ is the symmetrized gradient tensor, $d = \nabla \cdot \vec{u}$ is the divergence of the flow, and $\overset{\leftrightarrow}{I}$ is the unit tensor. The first scalar μ is the dynamic shear viscosity whereas λ associates with the bulk viscosity $\eta = 2\mu/3 + \lambda$.

Despite their deceivingly simple physical content, essentially mass and momentum conservation (Newton's law), as applied to a finite volume of fluid, the NS equations prove exceedingly difficult to solve as they involve the nonlinear evolution of a three-dimensional vector field, often in a complex geometry setup. This sets a formidable challenge to even the most advanced computational methods, whence the ceaseless hunt for more efficient numerical methods. Those methods include a broad array of techniques to discretize the NS equations, using finite differences, elements, and volume schemes.

Three decades ago, however, an entirely different route was devised, which consists of attacking fluid dynamics "from the bottom," i.e., appealing to a microscopic description of the fluid states matter, namely, a highly stylized version of molecular dynamics known as lattice gas cellular automata (Chopard and Droz, 1998; Rivet and Boon, 2005). In a nutshell, the idea is to introduce a boolean lattice fluid, consisting of a set of boolean particles whose dynamics is confined to the lattice sites. Boolean here means that the state of the system at a given lattice site and instant in time is uniquely defined by a set of b binary digits, coding for the absence or presence of a corresponding particle moving with unit speed along one of the b links connecting each lattice site to its neighbors. By a suitable choice of the lattice connectivity and interaction rules, such a boolean system can be shown to reproduce the NS equations of continuum fluid dynamics. An associated LB equation was also derived in the process of taking the boolean automaton to NS, but its computational capabilities went unnoticed. Even though the lattice gas cellular automata did not make it into a competitive tool for computational fluid dynamics, it nonetheless paved the way for the idea of computing fluid flows by simulating fictitious particle dynamics instead of discretizing the NS equations. The LB method (Higuera, Succi, and Benzi, 1989) fully inscribes within this line of thought.

C. Boltzmann equation for biology

The Boltzmann factor $e^{-E/k_B T}$ is a household name in biology as it governs the statistical behavior of a broad class of equilibrium and nonequilibrium (activated processes) phenomena of utmost relevance to biological systems. But, how about the Boltzmann equation? At first sight, the Boltzmann kinetic theory, in its original form at least, offers little scope for biological applications, since it formally applies to dilute states of matter in which molecular collisions are rare, the so-called weakly coupled regime, in which many-body interactions can safely be neglected with respect to binary molecular encounters. On the contrary, most biological phenomena are hosted primarily by condensed and soft-matter systems in which many-body effects play a primary role.

However, here a change of perspective proves exceedingly fruitful. Rather than the original Boltzmann equation for actual molecules, we refer to *model* Boltzmann's equations for *fluid particles*, the latter denoting the effective degrees of freedom describing the behavior of representative groups of molecules. Historically, the distinctive feature of model Boltzmann equations is a dramatic simplification of the collision operator in an attempt to relinquish most mathematical complexities while retaining the essential physics at hand. The most popular Boltzmann model is the celebrated Bhatnagar-Gross-Krook (BGK) equation, in which the collision operator is replaced by a simple single-time relaxation term (Bhatnagar, Gross, and Krook, 1954):

$$C_{\text{BGK}} = -\omega(f - f^{\text{eq}}), \tag{17}$$

where f^{eq} is the local equilibrium and ω is a relaxation frequency controlling the relaxation to the local equilibrium on a time scale $\tau = 1/\omega$.

The key advantage of this change is that model equations are more flexible by construction; hence they can be modified (extended) not only to simplify the collision operator but also to describe a wide host of physical effects not included in the original Boltzmann equation.

In particular, they can reinstate the effects of (i) many-body interactions, via effective one-body forces, in the spirit of density functional theory (the Vlasov-Boltzmann

equation) (Hansen and McDonald, 1990); (ii) statistical fluctuations through appropriate stochastic sources (the fluctuating Boltzmann equation) (Ladd, 1993); and (iii) far-from-equilibrium inhomogeneities, via suitably extended collision-relaxation operators in which the relaxation time is promoted to the status of a self-consistent dynamic field (Higuera, Succi, and Benzi, 1989; d'Humières, 1992; Chen *et al.*, 2003).

These three extensions inscribe to the general framework of the reverse kinetic theory (RKT), the strategy whereby the kinetic equation is designed top down, based on prescriptions securing compliance with macroscopic hydrodynamics in the first place, and then adding "molecular" details "on demand" by the specific application under investigation.

RKT reverses the canonical bottom-up route, whereby kinetic equations are derived from the underlying microscopic models and prove quite effective in bringing Boltzmann-like equations within the realm of condensed and soft-matter physics. However, care must be exercised in securing compliance of the top-down approach with the basic principles of statistical physics, namely, symmetries or conservation laws as well as evolutionary constraints (the second principle and its local form, known as the H theorem).

After this is all said and done, a practical question still remains: although simplified, the model BEs still leave in an unwieldy $(6+1)$-dimensional space. Here a time-honored ally of any computational scientist, the lattice, makes its entry.

D. Hydrodynamics for biology

Hydrodynamics and biology couple across an amazingly broad spectrum of scales, ranging from the macromolecular level in the compartments of living cells all the way up to industrial bioreactors, rivers, lakes, and oceans. From macromolecular motion in crowded cellular spaces to the deformations of membranes, to the motion of cells and the self-propulsion of bacteria, the common goal is to capture the effect of hydrodynamics under both equilibrium and nonequilibrium conditions. The latter may arise under the effect of an external flow, such as the transport of proteins and cells in the blood stream, or via the consumption of energy in cellular metabolic pathways. Indeed, the response of biological matter to temperature, pH, or mechanical forces plays a key role in most biological processes. For instance, biological macromolecules, cells or tissues, are usually fragile and easily damaged by hydrodynamic or shear forces as they occur far from equilibrium and under strong confinement. Following in the footsteps of hydrodynamic experimental techniques, customarily used to measure basic molecular properties, such as weight, size, and shape, computer simulations can be used in biophysical chemistry to explore and assess the mechanical and dynamical properties of macromolecules far from equilibrium, hence much closer to the *in vivo* conditions relevant to medical purposes. The benefits of including hydrodynamic forces is even more apparent whenever thermodynamic forces, stemming from solute-solvent interactions, can also be taken into account (see Fig. 5). A comprehensive computational framework including both hydrodynamic and thermodynamic forces stands therefore as a most desirable target. Owing to its inherently intermediate nature, between atomistic and
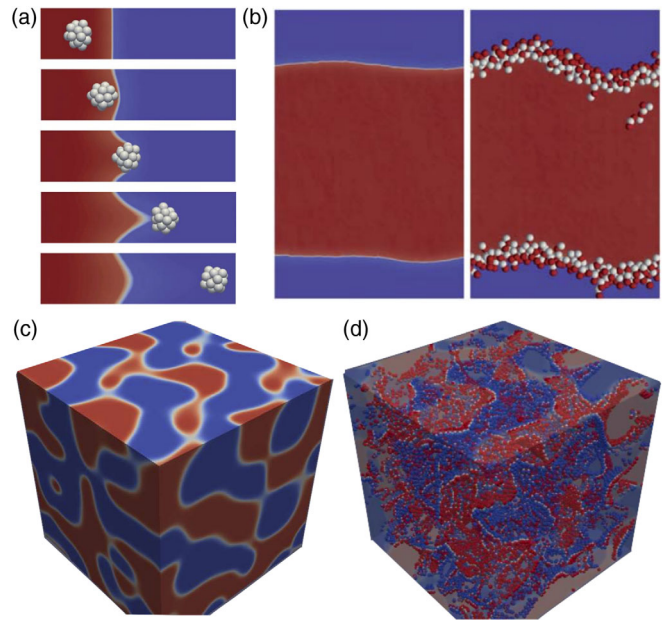
FIG. 5. Snapshot of molecules and particles in a multiphase flow, either (a) dragging the fluid from one phase to another or (b) sitting at the fluid-fluid interface. From Sega *et al.*, 2013. (c) In absence or (d) in the presence of colloids modifying the shape of the interface to a corrugated one to reflect molecular correlations. From Tiribocchi *et al.*, 2019.

continuum descriptions of matter, lattice kinetic theory sits at a vantage point to meet this goal.

III. LATTICE BOLTZMANN METHOD FOR CONTINUUM HYDRODYNAMICS

The basic idea of the LBM is to constrain the velocity degrees of freedom to a discrete lattice with sufficient symmetry to protect the conservation laws which secure the emergence of standard hydrodynamic behavior in the macroscopic limit. The LBM is based on the idea of representing fluid populations on a uniform Cartesian grid.

The standard LB scheme in single-relaxation time (BGK) form reads as follows (Higuera, Succi, and Benzi, 1989; Qian, d'Humières, and Lallemand, 1992):

$$f_i(\vec{r} + \vec{c}_i; t + \Delta t) - f_i(\vec{r}; t)$$
$$= -\omega[f_i(\vec{r}; t) - f_i^{\text{eq}}(\vec{r}; t)] + S_i(\vec{r}; t), \quad i = 0, b, \quad (18)$$

where f_i is the discrete Boltzmann distribution associated with the discrete velocity $\vec{c}_i$, $i = 0$, and b running over the discrete lattice to be detailed shortly. In Eq. (18), ω is a relaxation parameter controlling the fluid viscosity and f_i^{eq} is the lattice local equilibrium, basically the local Maxwell-Boltzmann distribution truncated to the second order in the Mach number. The truncation is not a luxury, but a requirement dictated by the Galilean invariance (GI).

Remember that GI refers to the invariance of the NS equations under an arbitrary change of the local fluid velocity $\vec{u} \rightarrow \vec{u}'$; upon such change, the NS equations stay the same, provided $\vec{u}$ is replaced by $\vec{u}'$. In continuum kinetic theory,

Galilean invariance is encoded within the dependence of the local Maxwell-Boltzmann distribution on the relative velocity of the molecules with respect to the fluid one, namely, $\vec{v} - \vec{u}$. As a result, an observer in the comoving frame, that is a frame moving at the local fluid velocity, experiences the same local equilibrium as if there were no fluid motion. Note that, in Eq. (18), $\vec{u} \equiv \vec{u}(\vec{r}, t)$ is an arbitrary function of space and time, indicating that Galilean invariance is a local and continuum symmetry, i.e., it holds even if different regions of the fluid move at different velocities, which is of course the case in most fluids of practical interest. Galilean invariance is reflected by the specific form taken by the moments of the equilibrium distribution, and specifically by those explicitly relevant to hydrodynamics, namely, mass, momentum, and the momentum flux tensor, that is,

$$\int f^{eq}\{1, v_a, v_a v_b\} d\vec{v} = \{\rho, \rho u_a, \rho u_a u_b + p\delta_{ab}\},$$

where the Latin subscripts run over spatial coordinates x, y, and z. This expression follows straight from the property of Gaussian integrals in velocity space.

One might naively expect that the same would be true in the lattice, provided the local Maxwell-Boltzmann distribution is retained with the plain replacement $\vec{v} = \vec{c}_i$. Straightforward algebra shows that the situation is different not for a mere mathematical accident, but as a consequence of the fact that a local and continuum symmetry cannot remain unbroken in a discrete lattice. More precisely, it cannot remain unbroken for any arbitrary velocity field. It turns out, though, that this is possible whenever the fluid velocity is much smaller than the sound speed, i.e., in the low-Mach number limit.

Under such a perturbative approximation, replacing velocity integrals with discrete sums returns exactly the same moments as in the continuum, namely,

$$\sum_i f_i^{eq}\{1, c_{ia} c_{ia} c_{ib}\} = \{\rho, \rho u_a, \rho u_a u_b + p\delta_{ab}\}, \quad (19)$$

provided that lattice tensors up to fourth order are isotropic. Of course, this by no means implies that the GI is fully restored, but simply that the GI-breaking terms are confined to kinetic moments higher than the order of 4.

In other words, the hydrodynamic constraints can be matched perturbatively by expanding the local Maxwellian to second order in the Mach number, which is sufficient to recover the isothermal NS equations, since those equations are quadratic in the fluid velocity. Full Galilean invariance for an arbitrary flow field u_a implies instead an *infinite* series in the Mach number, corresponding to the full expansion of the local Maxwellian in Hermite polynomials.

The actual expression of the discrete local equilibria reads as follows:

$$f_i^{eq} = w_i \rho (1 + u_i + \tfrac{1}{2} q_i), \quad (20)$$

where $u_i = c_{ia} u_a / c_s^2$ and $q_i = (c_{ia} c_{ib} - c_s^2 \delta_{ab}) u_a u_b / c_s^4$ represent the dipole and quadrupole contributions, respectively. Hereafter, repeated indices are summed upon. In Eq. (20), w_i is a set of lattice-dependent weights, normalized to unity, which represent the lattice analog of the global (no-flow) Maxwell-Boltzmann distribution. Finally $c_s^2 = \sum_i w_i c_{ia}^2$ is the lattice sound speed.

We hasten to note that, at variance with its continuum counterpart, Eq. (20), being a polynomial truncation of the Maxwell-Boltzmann equilibria, is non-negative definite only in a finite range of fluid velocities, typically of the order of $u/c_s \sim 0.3$. This configures the LB method as an appropriate description of quasi-incompressible, low Mach-number flows.

The discrete velocities matching isotropy constraints up to fourth order can be shown to be subsets of the D3Q27 (DnQm is a widely used notation to indicate a LB scheme in n dimensions using a set of m velocities) mother lattice with 27 discrete speeds in three spatial dimensions. D2Q9 and D3Q27 are the direct tensor product of the elementary one-dimensional D1Q3 stencil, $c_{ix} = \{-1, 0, +1\}$, in two and three spatial dimensions, respectively (see Fig. 6).

The expansion of Eq. (20) implies that hydrodynamic LB flows are bound to be quasi-incompressible, i.e., Mach number well below unity. Likewise, third-order kinetic moments, describing energy and heat flux, are not correctly reproduced, since these terms require sixth-order isotropic lattice tensors. Such constraints cannot be met by lattices confined to the first Brillouin region described by D3Q27: higher-order lattices extending beyond the first Brillouin cell are necessary. So much for the low Mach number
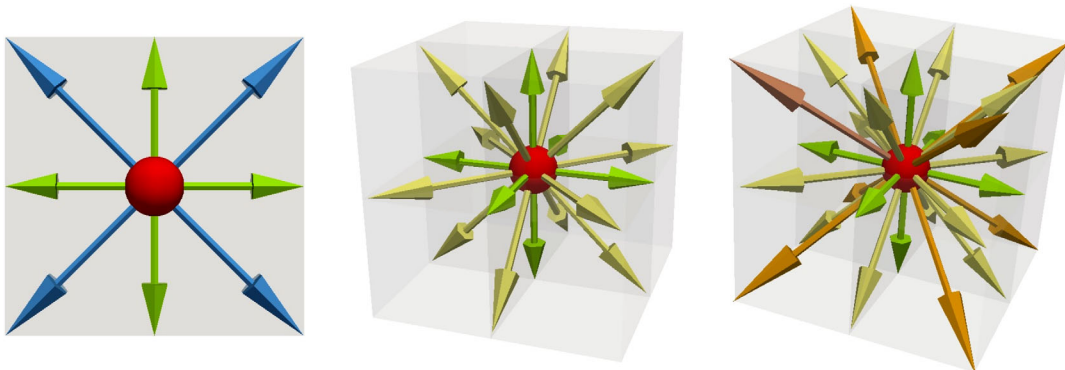

FIG. 6. Examples of standard 2D and 3D LB lattices, with 9, 19, and 27 discrete speeds, typically denoted as D2Q9, D3Q19, and D3Q27, respectively.

approximation, which is specific to the lattice. But what about the low-Knudsen limit, which, on the contrary, lies at the very roots of the convergence of Boltzmann to NS?

Since the Knudsen number controls the heterogeneity-driven departures from local equilibrium, it is intuitively clear that the low-Knudsen hydrodynamic limit implies further constraints on the nonequilibrium component of the momentum-flux tensor, which amounts to recovering the continuum expression of the stress tensor.

Based on Eqs. (19) and (20), it can be shown that in order for such constraints to be met, isotropic tensors, again up to the order of 4, need to be exactly reproduced in the lattice. It may come as a surprise that nonequilibrium constraints can be matched at the same order of isotropy of the equilibrium ones. The reason is parity invariance, namely, the fact that each discrete lattice velocity $\vec{c}_i$ comes with a mirror partner $\vec{c}_{i^*} = -\vec{c}_i$. Indeed, nonequilibrium constraints involve lattice corrections driven by fifth-order lattice tensors, which are identically zero due to parity invariance. Readers interested in the straightforward but laborious algebraic details can look up the vast literature on the subject (Hénon, 1987; Chopard and Droz, 1998; Rivet and Boon, 2005).

Next, let us comment on the source term S_i on the right-hand side of the LB equation (18). This term stands for a generic source of mass, momentum, and energy, describing the coupling of the fluid to the surrounding environment. Mass sources are typically associated with the presence of chemical reactions, turning one species into another in multicomponent versions of the LB for reactive flows.

In the case of inert flows, the source term typically encodes the momentum exchange due to the coupling to external (or internal) fields, such as gravity or more complex interactions, like self-consistent forces reflecting potential energy interactions within the fluid, as well as thermal fluctuations. As we shall see, the two latter cases are crucial for the extension of the LB method beyond NS hydrodynamics.

From the operational standpoint, the source term S_i acts like a bias promoting the populations which move along the local force field and setting a penalty on those that move against it. It is therefore clear that the strength of such term is subject to stringent stability and positivity constraints.

A. From the lattice Boltzmann method to continuum hydrodynamics

The conceptual path taking the LBM to NS is exactly the same as in the continuum theory, with the crucial caveat of turning around the many catches associated with lattice discreteness. To make a long story short, it amounts to securing the proper symmetries of the lattice tensors entering the set of lattice moment equations. As always with lattice physics, the name of the game is to erase the lattice dependence to the desired order. For the case of isothermal, incompressible fluids, the desired order is the *fourth* one. In equations, and using coordinate notation for tensors (Hénon, 1987),

$$\sum_i w_i = 1, \qquad (21)$$

$$\sum_i w_i c_{ia} c_{ib} = c_s^2 \delta_{ab}, \qquad (22)$$

$$\sum_i w_i c_{ia} c_{ib} c_{ic} c_{id} = c_s^4 (\delta_{ab}\delta_{cd} + \delta_{ac}\delta_{bd} + \delta_{ad}\delta_{bc}), \qquad (23)$$

where the Latin indices run over spatial coordinates. In Eqs. (21)–(23), w_i is a set of weights normalized to unity and odd-rank tensors are automatically equal to zero by parity invariance, i.e., every discrete velocity c_{ia} comes with an equal and opposite partner, so that the sum of the two gives a null. Finally c_s^2 is a lattice-dependent constant expressing the sound speed in the lattice, as per the expressions (21)–(23).

With the symmetries secured, everything proceeds as in the continuum theory, with another important caveat though, namely, the fact that the effective mean-free path of the lattice fluid is replaced by the lattice spacing Δx whenever the latter is larger than the physical one, the typical case in most macroscopic LB simulations.

This is a very technical and somewhat thorny issue, whose details are out of the scope of the present review. Nevertheless, we caution the reader that the physics taking place at the scale of a few lattice spacings should always be inspected with a large pinch of salt, because it is constantly in danger of breaking the hydrodynamic assumptions.

Once these catches are disposed of, one ends up with a lattice fluid obeying an ideal equation of state $p = \rho c_s^2$ and a kinematic viscosity given by

$$\nu = \left(\frac{1}{\omega} - \frac{1}{2}\right)\nu_l, \qquad (24)$$

where $\nu_l = c_s^2 \Delta x^2 / \Delta t$ is the natural lattice viscosity. Note that the stability range of the discrete time marching $0 < \omega < 2$ also secures the positivity of the kinematic viscosity.

Both ends of this range must be handled with care. In the low-viscosity regime $\omega \to 2$, typical of turbulence, strong gradients may develop posing a serious threat to the numerical stability of the scheme. A powerful variant of the basic LB method includes a self-consistent tuning of the relaxation parameter ω so as to ensure compliance with local entropy growth (H theorem) (Karlin, Ferrante, and Öttinger, 1999; Succi, Karlin, and Chen, 2002). That variant, known as the entropic lattice Boltzmann (ELB) method, is normally intended to simulate high-Reynolds macroscopic flows, but lately is proving very effective also to stabilize microscale LB simulations with sharp interfaces.

In the opposite high-viscous regime $\omega \to 0$, the one most relevant to biological applications, the viscosity may formally diverge, signaling a departure from hydrodynamic behavior, due to the onset of ballistic motion. As a result, whenever possible, LB simulations should be kept away from both limits, say $\nu_{\mathrm{LB}} \sim 0.1$. We return to this important point in Sec. VII.

B. Boundary conditions

In the early days, boundary conditions were hailed as one of the main assets of the LB method, and, to a certain extent, they still are. As a matter of fact, since information always travels
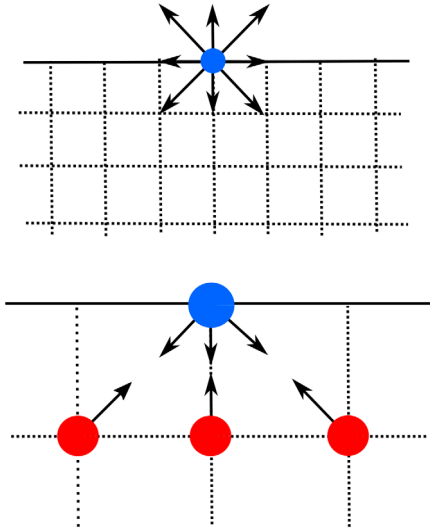
FIG. 7. The effect of wall boundaries on the LB populations according to the bounce-back scheme. Populations hitting a wall node are reflected from the incoming fluid direction, giving rise to no-slip flow conditions.

along straight lines, even complex geometries can be handled by comparatively straightforward computational methods based on elementary mechanical operations. For instance, no slip on solid walls can be implemented through a simple bounce back between distributions propagating along opposite directions, i.e., from the fluid to the wall and vice versa; see Fig. 7.

For generality, let us consider a fluid flowing in a bounded domain Ω confined by a surrounding boundary $\partial\Omega$. The problem of formulating boundary conditions within the LBE formalism consists of finding an appropriate relation expressing the incoming (unknown) populations $f_i^<$ as a function of the outgoing (known) ones $f_i^>$. Outgoing populations at a boundary site $\vec{x}$ are defined by

$$\vec{c}_i \cdot \vec{n} > 0,$$

where $\vec{n}$ is the outward normal to the boundary cell centered in $\vec{x}$. Incoming populations are defined by the opposite sign of the inequality. In mathematical terms, this relationship translates into a linear integral equation

$$f_i^<(\vec{x}) = \sum_y \sum_j B_{ij}(\vec{x} - \vec{y}) f_j^>(\vec{y}), \qquad (25)$$

where the kernel $B_{ij}(\vec{x} - \vec{y})$ of the boundary operator generally extends over a finite range of values $\vec{y}$ inside the fluid domain. This boundary operator reflects the interaction among the fluid molecules and the molecules in the solid wall. Consistently with this molecular picture, boundary conditions can be viewed as a special (sometimes even simplified) type of collisions between fluid and solid molecules. Physical fidelity can make the boundary kernel quite complicated, which is generally not the idea with LBE. Instead, one usually looks for expressions minimizing the mathematical burden without compromising the essential physics.

In particular, one seeks minimal kernels fulfilling the desired constraints on the macroscopic variables (density, speed, temperature, and possibly the associated fluxes as well) at the boundary sites $\vec{x}$. This may lead to a mathematically underdetermined problem, more unknowns than constraints, opening up an appealing opportunity to accommodate more interface physics within the formulation of the boundary conditions. However, it also calls for some caution to guard against mathematical ill posedness. By now, vast literature which, however, goes beyond the scope of the present review is available on this topic (Chen, Martinez, and Mei, 1996; Zou and He, 1997; Krüger *et al.*, 2017).

Here we point out that the main issue controlling the complexity of the boundary problem is whether the collection of boundary points lies on a surface aligned with the grid (so-called staircase wall boundary) or it is given by a set of off-grid surface elements (sometimes called surfels). The latter case is significantly more complex and requires extra care, the immediate benefit being that the near-wall physics is second-order accurate as compared to the first-order accuracy of the staircased approximation [for a detailed account see Krüger *et al.* (2017) and Succi (2018)]. Advanced applications of the off-lattice boundary method prove capable of dealing with fairly complex geometries, an important but highly technical topic which also goes beyond the scope of the present review. For full details see the recent books by Krüger *et al.* (2017) and Succi (2018).

C. The bright sides of the lattice Boltzmann method

Why does the LB method represent an appealing scheme for simulating complex states of flowing matter? Several features merit highlight, but essentially they all come down to the benefits of working with extra dimensions opened up by the six-dimensional phase space inhabited by kinetic theory.

More in detail, the upshots are the following: First, the information always travels along straight lines, regardless of the space-time complexity of the emergent hydrodynamics. What we mean by this is that the discrete distributions move in sync along the discrete light cones, defined by

$$\Delta\vec{r}_i = \vec{c}_i \Delta t,$$

no matter the complexity of their space-time dependence.

Since the discrete velocities do not depend on space and time, the streaming is exact, no information is lost in hopping from one lattice site to another, literally an error-free operation. This stands in sharp contrast with the self-advection term $\vec{u} \cdot \nabla\vec{u}$ of macroscopic hydrodynamics, whereby the velocity fields, or any other field for that matter, moves along space-time dependent material lines defined by the fluid velocity itself, typically a complicated function of space and time. Such independence explains the outstanding LB amenability to parallel computing, a practical asset which cannot be underestimated.

Second, space and time always come on the same (first-order) footing. In particular, this means that dissipation is not expressed by second-order spatial derivatives (Laplace operator) but simply as a local relaxation to a local equilibrium, as described earlier. This is a substantial advantage for confined

flows, whose dynamics is largely dictated by the presence of solid boundaries, where spatial gradients are usually at their largest. Taking space derivatives near the boundaries is notoriously prone to numerical inaccuracies, especially whenever the geometry of the domain is irregular, as is often the case for biological flows. LB equipped with suitable formulations for curved boundaries can significantly mitigate such difficulties.

Third, LB carries pressure as just any other macroscopic field, with no need of solving a usually expensive Poisson problem to compute the pressure field consistent with an incompressible flow. This is because the discrete distributions carry the momentum-flux tensor "on their back" and consequently the pressure obeys its own dynamic equation. More specifically, the second-order momentum-flux tensor P_{ab} obeys the relaxation equation

$$\partial_t P_{ab} + \partial_c Q_{abc} = -\omega(P_{ab} - P_{ab}^{eq}), \qquad (26)$$

where $Q_{abc} = \sum_i f_i c_{ia} c_{ib} c_{ic}$ is the third-order energy-flux tensor, and Latin indices are running over spatial dimensions. In the limit where the P_{ab} tensor is enslaved to its equilibrium value, the time derivative can be dropped and the energy-flux tensor can be approximated by its equilibrium expression. Under such conditions, Eq. (26) delivers $P_{ab} \sim P_{ab}^{eq} + \omega^{-1}\partial_c Q_{abc}^{eq}$.

Once the due lattice symmetries are fulfilled, the equilibrium component delivers the advection and pressure terms of the NS equation, while the third-order nonequilibrium term delivers the dissipative term.

Fourth, coupling of the fluid to a broad variety of other physical phenomena is readily achieved by formulating suitable expressions of the source term S_i.

Typically this reflects the action of mesoscale forces describing the effective interactions between fluid-fluid and fluid-solid molecules. The positive side effect is that an entire new world of complexity may be supported simply by inserting a few lines of additional code.

This feature is crucial to the extension of the LB method beyond the traditional realm of dilute gases and particularly to biological flows as we detail in the sequel.

Once again, this rosy picture refers only to the conceptual fresco, actual implementations requiring great care in circumventing lattice artifacts.

D. The dark side of the LB moon

Following Frenkel, it is often more instructive to analyze what can go wrong in a computer simulation, rather than indulging in the description of its success stories. To abide by this, in the following we mention things that can still go wrong with LB simulations, the dark side of the LB moon.

The LB method draws much of its computational simplicity from the features of uniform and regular lattices that, at times, recall an ideal Legoland. Realistically complex geometries may sometimes challenge this setup and call for more flexible and elaborate formulations, such as interpolation procedures for curved boundaries, local grid refinement, or mergers with finite-volume techniques, to name just a few of some popular options. It is only fair to acknowledge that implementing such

extensions may be laborious and possibly tax the computational simplicity as well.

Another limitation is the finite compressibility. While buying major computational savings through dispensing with the Poisson problem for pressure, a finite amount of compressibility must be tolerated in return. Such effects are usually negligible for low Reynolds flows, but they need nevertheless to be watched carefully in open flows, especially at outlets, where spurious density waves may eventually backpropagate into the fluid and undermine the accuracy and sometimes even the stability of the simulation.

A similar observation applies to flows with strong thermal transport. For one, it should be appreciated that the LB method is essentially *athermal* in the sense that the discrete Boltzmann distribution is represented by a linear superposition of monochromatic beams with no dispersion in velocity space, hence, zero temperature in a strict equilibrium thermodynamic sense. Nevertheless, a kinetic temperature can still be defined as a measure of the dispersion around the mean flow velocity, namely, in D spatial dimensions $Dk_BT = \sum_i f_i(\vec{c}_i - \vec{u})^2$. The inclusion of strong thermal effects typically requires higher-order lattices, accommodating sixth-order tensors describing the flux of energy (Nie, Shan, and Chen, 2008; Krüger *et al.*, 2017). Several alternatives are also possible which go beyond the scope of the present work [for details see Chapter 22 in Succi (2018)]. Despite major progress, it is fair to say that thermal LB schemes still lag behind their athermal counterparts in terms of numerical robustness. Consequently, the LB method is often coupled to independent thermal solvers typically based on finite-difference or finite-volume methods.

These limitations said, we believe it is fair to surmise that the appealing features of the LB method largely outweigh its weaknesses.

IV. LATTICE BOLTZMANN FOR GENERALIZED HYDRODYNAMICS

The ideas and methods presented so far deal with flows of "simple" fluids that can be described in terms of the NS equations (note that simple fluids can give rise to highly *complex* flows, turbulence being a prime example in point).

Modern high-tech applications, nanoengineering and biology in the first place, set a pressing demand of quantitative understanding of more complex states of flowing matter, in which fluids interact with external or self-consistent fields, undergo chemical reactions, phase transitions, or interact with a variety of suspended bodies, such as colloids or biological molecules.

This emerging sector of modern science, often referred to as "complex fluids" or more trendily "soft-matter physics" portrays a multidisciplinary scenario whereby fluid dynamics makes contact with other disciplines, primarily chemistry, material sciences, and biology as well.

There is growing evidence that the LB method, and extensions thereof, holds a vantage point as a computational framework for the simulation of complex states of flowing matter. Ideally, the LB method would fill the gap between fluid dynamics and molecular dynamics, namely, the large and all-important region where fluid dynamics breaks down and molecular dynamics is not yet ready to take over for lack of
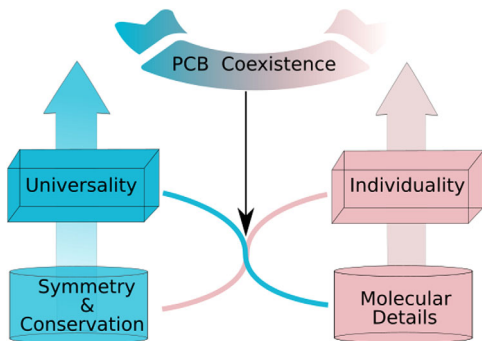
FIG. 8. The description of physical-chemical-biological (PCB) systems requires a suitable integration of microscopic details (individuality) within the universal harness dictated by symmetry principles and ensuing conservation laws which govern the macroscopic behavior (universality). Kinetic theory is expected to strike an optimal (problem-dependent) balance between universality and individuality.

efficient algorithms and computing power; see Fig. 8 (Boon and Yip, 1991).

The LB method is a natural candidate to fill that gap because of its mesoscopic ability to incorporate microscopic details into the kinetic theory formalism via suitable external fields and/or equivalent generalizations of local hydrodynamic equilibria: a microscope for fluid mechanics, a telescope for molecular dynamics (Succi, 2018).

The extension of the LB method to generalized flows is based on a number of major upgrades of the basic LB theory. In the sequel, we focus on the following selections:
- reactive systems,
- charged flows,
- flows far from equilibrium,
- fluctuating hydrodynamics,
- nonideal fluids, and
- flows with suspended bodies.

We now proceed to a more detailed discussion of these items.

A. Advection-diffusion-reaction systems

Chemical reactivity is an essential element to deal with when facing the task of simulating systems at the PCB interface (Boon *et al.*, 1996; Coveney, Boon, and Succi, 2016; Alowayyed *et al.*, 2017; Succi, 2018).

Biochemical reactions lie at the heart of most biological phenomena, controlling species interconversion. They involve the breaking and making of covalent bonds, often catalyzed by enzymes, ubiquitous in all metabolic and synthetic reactions within the cell and in the body. Reactions occur in bulk conditions in single-phase (homogeneous) or multiphase (heterogeneous) environments, typically at the interface between phases or in a porouslike environment. Biological reactions take place under a wide host of different conditions. The antigen-antibody binding in solution or the binding of small molecules to plasma proteins in blood, clotting reactions on the surface of blood vessels, the hydrolysis of adenosine triphosphate (ATP) on the surface of endothelium, ion transport, and the release of nitric oxide are examples of homogeneous reactions. Heterogeneous reactions within tissues include aerobic metabolism, glucose consumption, and receptor-mediated endocytosis.

In all cases, reactions can occur in either no-flow or underflow conditions, as, for example, the enzyme reactions on the surface of the blood vessels. Diffusion and convection affect the rates of homogeneous and heterogeneous reactions. Because reactants must diffuse or convect to the surface where they react, the mass transfer mechanisms occur in sequence with the reaction process. In many cases, the effects of reaction and diffusion on the reaction rates cannot be easily separated and numerical methods provide the only viable route to study the process.

The LB method is a powerful framework to capture diffusion and reaction since its mathematical apparatus is by no means confined to fluid equations. As a matter of fact, by suitable tuning of the local equilibria, relaxation matrix, and external forces, it permits one to generate a broad variety of linear and nonlinear partial differential equations, including those describing relativistic and nonrelativistic quantum phenomena (Succi, 2002a; Mendoza *et al.*, 2010).

Of relevance are advection-diffusion-reaction (ADR) equations of the general form

$$\partial_t C + \vec{u} \cdot \nabla C = D\Delta C + R(C), \qquad (27)$$

where C is a species concentration, say molecules per unit mass or volume. The left-hand side describes the passive transport along the fluid material lines, whereas the right-hand side describes diffusion plus the effects of chemical reactions occurring in the fluid moving at the barycentric velocity $\vec{u}$.

Although thermodynamically favorable, many reactions are limited by the energy barrier which needs to be crossed in order to form the activated state, so that, in the absence of the catalyst, reactions simply do not occur. In its presence, the rate of reaction can increase dramatically, although the change in energy between reactants and products is not affected. In fact, the enzymes affect the rate of a reaction, not its equilibrium. For many reactions involving a single substrate, the rate of consumption of the substrate follows the Michaelis-Menten equation

$$R(C) = \frac{RC}{K + C},$$

where R is the magnitude of the rate of disappearance of the substrate (or reactant) and K is the Michaelis constant, the concentration of the substrate at which the reaction attains half of its maximum rate.

Another reaction term particularly relevant to populations biology is the logistic expression

$$R(C) = aC - bC^2,$$

where a describes the Malthusian growth rate and b characterizes the nonlinear decay of the species due to competition for, say, space and/or food. The ADR class falls naturally within the formalism, by simply defining a LB distribution C_i for the concentration, such that $\sum_i C_i = C$ and specifying the following local equilibria:

$$C_i^{\text{eq}} = w_i C \left(1 + \frac{\vec{u} \cdot \vec{c}_i}{c_s^2}\right),$$

where both fluid and lattice speeds are normalized by the sound speed c_s. To be noted that in the above $\vec{u}$ is the fluid speed, which does not match the current $C\vec{u} \neq \sum_i C_i \vec{c}_i$, because ADRs conserve mass but not momentum. The LB method is then particularly well suited to solve the ADR equations in complex geometries, such as those occurring in morphogenesis (Ponce Dawson, Chen, and Doolen, 1993; Ayodele, Varnik, and Raabe, 2011), heterogeneous catalysis (Falcucci *et al.*, 2016), and related phenomena.

B. Charged fluids

Electrostatics plays a vital role in biological processes and requires handling electrolytic solutions and charged solutes typically in flow conditions, the realm of the so-called electrokinetics. Solutes range from strongly charged molecules, such as DNA, to weakly charged macromolecules, such as proteins, where partially screened mobile charges and charged surfaces are ubiquitous conditions of biological settings. Prominent examples include (i) the stability of proteins as a function of pH and ionic strength, such as salting-in and salting-out effects due to the interplay of protein charges with the aqueous or saline environment; (ii) protein-ligand association processes, including enzymatic allostery modulated by salt bridges, salt bridges in virus assemblies, and thermal stability; and (iii) membrane proteins and their specific electrostatic properties, whereby the collective charges of the intracellular residues tend to be more positive as compared to the extracellular ones (the so-called "positive inside" rule), and so on. The associated biological flows are driven by electro-osmosis and electrophoresis. Well-known examples of those processes include (i) ion channels and the gating of ions across the cell membrane that regulate electrical signaling in secretory and epithelial cells, as much as the cell volume; (ii) the electrokinetic flow due to the glycocalyx layer covering cells, a polyelectrolyte exhibiting a surface charge and responsible for an electrochemical gradient that regulates mechanotransduction; and (iii) the mitochondria and the chloraplast, where proton gradients generate a chemiosmotic potential, also known as a proton motive force, for the synthesis of ATP.

Although the general principles of electrokinetics are fairly well understood (Masliyah and Bhattacharjee, 2006), methods that translate these principles into accurate numerical predictions are still in their infancy. In fact, deriving the interactions between charged solutes and the solvent requires computing the interactions of a large number of molecules and the averaging of these over many solvent configurations. Such daunting computational requirements are partly overcome through the continuum mesoscopic approach, whereby solvent and ions are described in the continuum and in a preaveraged sense (Capuani, Pagonabarraga, and Frenkel, 2004; Reboux *et al.*, 2006; Wang and Kang, 2010).

The LB framework solves complex electrokinetic problems through an efficient formulation of the electrolytic solution as a multispecies problem: one species for the neutral solvent, water, and two for the positively and negatively charged ionic

components. The Poisson equation provides the solution for electrostatics and the self-consistent forces for the transport of the fluid species in the so-called Vlasov-Poisson approximation.

Let us briefly survey the LB method for charged multicomponent systems. The fluid mixture is composed of three sets of populations labeled by index $\alpha = 0$, 1, 2, two ionic components with charges $z_\alpha e$, e being the proton charge, density n^α, and velocity $\vec{u}^\alpha$. Given the barycentric velocity

$$\vec{u} = \frac{\sum_\alpha n^\alpha \vec{u}^\alpha}{\sum_\alpha n^\alpha},$$

the species relative velocity is $\delta \vec{u}^\alpha = (\vec{u}^\alpha - \vec{u})$. The aqueous medium accommodates solute biomolecules, with the ith particle having position $\vec{r}_i$ and valence z_i. The electrostatic potential is obtained by solving the Poisson equation,

$$\nabla^2 \psi = -\frac{1}{\epsilon} e \left[n^+ - n^- + \sum_n z_n \delta(\vec{r}_n - \vec{r})\right], \quad (28)$$

where the medium has dielectric permittivity ϵ, duly complemented by boundary conditions of the Dirichlet or Neumann kind. For insulating confining walls, where the surface has local charge density Σ, this reads $-\nabla \phi \cdot \hat{n} = \Sigma/\epsilon$, where $\hat{n}$ is the unit vector normal to the surface.

The dynamics of each species follows the evolution equation:

$$f_i^\alpha(r + c_i, t + 1) = f_i^\alpha(r, t) + \omega(f_i^{\alpha,\text{eq}} - f_i^\alpha) + S_i^\alpha, \quad (29)$$

where the Maxwellian equilibrium for mixtures is given by (Marconi and Melchionna, 2011a, 2011b)

$$f_i^{\alpha,\text{eq}} = w_i n^\alpha \left[1 + \frac{\delta \vec{u}^\alpha \cdot \vec{c}_i}{v_T^2} + \frac{[(\delta \vec{u}^\alpha \cdot \vec{c}_i)^2 - v_T^2(\delta u^\alpha)^2]}{2 v_T^4}\right] \quad (30)$$

and the force term

$$S_i^\alpha = w_i n^\alpha \left[\frac{\vec{F}^\alpha \cdot \vec{c}_i}{v_T^2} + \frac{[(\vec{c}_i \cdot \vec{u})(\vec{c}_i \cdot \vec{F}^\alpha) - v_T^2 \vec{F}^\alpha \cdot \vec{u}]}{v_T^4}\right].$$

The local self-consistent forces are $F^\alpha = F^{\alpha,\text{drag}} - ez^\alpha \nabla \psi$, where $F^{\alpha,\text{drag}} = -\omega_{\text{drag}}^\alpha \sum_\beta (n^\beta/n)(u^\alpha - u^\beta)$ is the drag force exerted on species α, resulting in a cross-diffusion coefficient $D^\alpha = v_T^2/\omega_{\text{drag}}^\alpha$, where $\omega_{\text{drag}}^\alpha$ is the relaxation frequency.

So much for the governing equations. However, a hallmark of charged systems is that local electrostatic forces can be dramatically intense, often exhibiting rapid spatial modulations of the electrolytic densities, as in the presence of double layers.

Such an occurrence can lead to severe numerical instabilities in methods based on the direct solution of the NS equation, as a consequence of local violations of the Courant-Friedrichs-Lewy stability condition (Courant, Friedrichs, and Lewy, 1928). Thanks to its inherently small time step (in macroscale units), the LB method can often handle stiff forces without losing stability. However, strong

polyelectrolytes are problematic to handle and cannot be treated directly, thus some sort of charge rescaling is required (Datar *et al.*, 2017). Despite these liabilities, LB simulations of electrolytic systems show stable behavior over a wide range of solute charges and molarities, in particular, in the submolar range that covers a large portion of biological conditions.

C. Flows far from equilibrium

Most biological systems operate far from equilibrium, i.e., they draw energy from their environment and dissipate heat back to it, thereby lowering their own entropy content at the expense of the environment. This is the operating principle of the so-called "dissipative structures," a cornerstone of nonequilibrium thermodynamics (Prigogine, 2017). In the process of dumping entropy to the environment, they manage to migrate across a sequence of different nonequilibrium steady states (NESS), in which they deliver different functions.

Examples of NESS in biology include molecular machines, cells in motion, metabolic pathways, and many others. For instance, the kinesin protein walks along the microtubule, carrying cargos from one part of the cell to another, by absorbing energy from ATP hydrolysis and converting chemical energy into mechanical work, of which ∼60% is used for motion and the rest is dissipated to the surroundings. Off-equilibrium conditions imply that biological agents exchange mass, momentum, energy, and entropy with the environment through transient or steady currents and fluxes. Clearly, such a complex space-time dependent network of currents and fluxes, typically operating on a broad spectrum of concurrent space and time scales, needs to be captured by the numerical approach (Takahashi, Arjunan, and Tomita, 2005), hence a major need for multiscale methods.

Indeed, these currents and fluxes typically occur through thin (atomic scale) interfaces, which resolve the tension between competing mechanisms, say chemical reactions and molecular diffusion, through a sudden spatial transition between distinct bulk phases, say the liquid and vapor in a multiphase flow.

For the case of a diffusion-reaction system, the width of the transition region can be estimated as

$$w \sim \sqrt{D\tau_{\mathrm{ch}}},$$

where D is the diffusion coefficient and τ_{ch} is a typical chemical time scale. By expressing $D \sim \lambda^2/\tau_c$, with λ being the mean-free path and τ_c the collisional time scale, we obtain $w \sim \lambda(\tau_{\mathrm{ch}}/\tau_c)^{1/2}$. This shows that, unless chemical reactions are much slower than inert collisions (slow chemistry), the interface width is comparable with the molecular mean-free path, or shorter (fast chemistry).

Another example in point is foams and emulsions, i.e., droplets (bubbles) of liquid (vapor) dispersed in a continuum liquid phase, typically water. In these multiphase systems, the transition between the dense and light phases is controlled by the surface tension which is, in turn, dictated by molecular interactions, notably the strength of the potential and its spatial range. That leads to interface widths of the order of the spatial
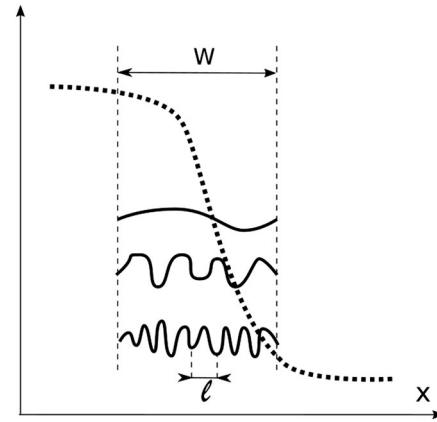

FIG. 9. A sketch of a high-density–low-density diffuse interface of width w, as typical for many biosystems (dotted line), for different values of the mean-free path, of length ℓ. Each solid line shows a different profile of the interface, depending on its intrinsic structure. Here we assume that the wavelength of the density profile coincides with ℓ. In the limit $\ell/w \ll 1$ (solid line, bottom), nonequilibrium effects are negligible and hydrodynamics holds. On the other hand, when ℓ becomes comparable to the interface width w (solid line, top), nonequilibrium effects can no longer be neglected and higher-order kinetic moments must be accounted for.

range of such interactions, typically nanometers or below. A typical estimate of the interface width w is as follows:

$$w \sim \sqrt{\frac{k_B T}{\sigma}},$$

where σ denotes the surface tension. Typical values in International System of Units are $k_B T \sim 4 \times 10^{-21}$ and $\sigma \sim 7 \times 10^{-2}$, deliver $w \sim 4 \times 10^{-10}$, i.e., a fraction of nanometer.

As shown in Fig. 9, such nanometric interfaces challenge the low Knudsen assumption which lays at the foundation of the hydrodynamic description. In fact, given that the interface thickness is comparable with the molecular mean-free path, the results are local Knudsen numbers of the order unity. In the last decade, a number of technical extensions of the original LB method have been developed with the aim of gaining insight into these complex nonequilibrium interfacial phenomena (and yet, much remains to be done).

Here we summarize the main upgrades, namely, (i) higher-order lattices, (ii) kinetic boundary conditions, and (iii) regularization techniques.

1. Higher-order lattices

As discussed in the initial part of this paper, hydrodynamics represents the "infrared" limit of kinetic theory, whereby all macroscopic heterogeneities live on much longer scales than the molecular ones (hydrodynamic transport regime). From a formal viewpoint, this means that the Boltzmann distribution is accurately described by its lower-order moments, typically density (order of 0), current (order of 1), and momentum flux (order of 2).

In the hydrodynamic transport regime, all higher-order moments (nonequilibrium excitations) are directly enslaved

to the low-order ones, hence they have no independent dynamics of their own.

Far from equilibrium, where strong inhomogeneities may persist down to near-molecular scales, such a low-order picture breaks down, and more kinetic moments concur to define the motion of matter beyond the hydrodynamic regime.

Without giving details, it is intuitively clear that this far-from-equilibrium regime requires the use of higher-order lattices, securing the proper recovery of the correspondingly higher-order moments. A formal theory of the LB method beyond NS equations, based upon higher-order lattices, was given by Shan, Yuan, and Chen (2006). They developed the discrete analog of Grad's expansion in Hermite polynomials with full details on its specific implementation on a series of higher-order lattices associated with different numerical quadrature rules. The first observation as compared to Grad's 13 moment formulation is that the scheme provides a larger set of degrees of freedom (Grad, 1949). The discrete speeds of the corresponding lattices are typically of the order of 40 or more, hence many more than the 13 Grad's moments (Zhang, Shan, and Chen, 2006; Meng and Zhang, 2011).

2. Kinetic boundary conditions

Higher-order lattices offer room for extra moments, but this is not sufficient *per se*, unless suitable boundary conditions are formulated near solid boundaries. This is where the lattice formulation makes a distinct contribution: while it appears hard to devise well-posed boundary conditions for the kinetic moments, in fact complicated nonlinear tensors, lattice formulations lend themselves to conceptually transparent and mathematically well-posed formulations. The reason is always the same: the information moves along straight lines and boundary conditions can be formulated in terms of mechanical relations between the outgoing (fluid-to-wall) and the incoming (wall-to-fluid) discrete distributions. Nonequilibrium flows exchange momentum with the solid walls along both tangential and normal directions: describing this exchange is the mandate of kinetic boundary conditions. Empirical forms adapting this constraint to the lattice environment have been formulated in terms of accommodation coefficients (Succi, 2002b). Following Maxwell (1878), the idea is that molecules impinging on the wall lose track of their incoming speed. Consequently, they are reinjected into the fluid along a random direction, with a velocity drawn from a Maxwellian at the local wall speed and temperature. Albeit handy, these accommodation schemes remain empirical in nature. A more satisfactory formulation was developed by Ansumali and Karlin (2002), who basically expressed the accommodation coefficients in terms of the outgoing (fluid-to-wall) distribution functions and their equilibrium version, thus providing a closed and consistent recipe. The Ansumali-Karlin boundary conditions exhibit a number of appealing properties. First, they preserve the positivity of the distribution at boundary nodes, i.e., if the incident distribution is positive, the reflected one is guaranteed to be positive too. Second, they readily extend to general wall scattering kernels, such as those allowing a blend of slip and reflection. At the time of this writing, these kinetic boundary conditions are the tool of the trade for finite Knudsen LB simulations.

3. Regularization

Regularization is a general and powerful idea across many fields of theoretical physics to remove various forms of divergences and singularities which arise whenever a given description or theory fails to capture the physics in point.

Kinetic theory is no exception. Indeed, it has been long known that posthydrodynamic equations beyond the NS level suffer a number of problems, primarily short-scale linear instabilities (Bobylev, 1982). Many regularization schemes have been proposed ever since to tame such instabilities (Struchtrup and Torrilhon, 2003, 2007). Regularization procedures have been (re)discovered only recently by the LB community (Latt and Chopard, 2006; Montessori *et al.*, 2014) and it is not yet clear how they relate to the corresponding counterparts in continuum kinetic theory. LB regularization consists of filtering out the contribution of nonhydrodynamic moments (ghosts) to the hydrodynamic ones: mass, momentum, and momentum flux.

Let us dig a little bit deeper into the subject. The standard LB scheme in BGK form reads (time step made unit for simplicity)

$$f_i(\vec{r} + \vec{c}_i; t + 1) = (1 - \omega)f_i(\vec{r}; t) + \omega f_i^{\text{eq}}(\vec{r}; t). \quad (31)$$

The actual distribution can be split as follows:

$$f_i = h_i + g_i,$$

where the hydrocomponent h_i collects terms up to third-order Hermite polynomials associated with mass, momentum, momentum flux, and energy flux, while g_i collects all higher-order terms, transport plus genuinely kinetic fields with no immediate macroscopic interpretation (ghosts in LB jargon). Formally, one defines a regularization operator $\mathcal{R}$, projecting the actual distribution onto the hydrodynamic subspace, i.e., $h_i = \mathcal{R}f_i$, that is $\mathcal{R}g_i = 0$.

Both hydro and ghost terms further split into equilibrium and nonequilibrium components, the ghost equilibrium being identically null by construction. Thus, what remains to be filtered out is just the ghost nonequilibrium, which is constantly revived at each free-streaming step, the nonequilibrium engine.

By applying the regularization projector to the right-hand side of the BGK equation, an operation corresponding to filtering out ghost components after streaming, we obtain the following regularized LB equation:

$$f_i(\vec{r} + \vec{c}_i; t + 1) = (1 - \omega)h_i(\vec{r}, t) + \omega f_i^{\text{eq}}(\vec{r}, t). \quad (32)$$

The regularized LB equation has recently provided significant benefits in terms of improving the stability of the LB scheme under finite Knudsen conditions.

The "posthydro" LB literature is vast and growing, but not conclusive yet. In particular, it is not clear how the three main ingredients mentioned should be combined in order to obtain correct finite Knudsen behavior.

A valuable step in the comprehension of LB posthydro capabilities was provided by Ansumali *et al.* (2007), who discovered exact solutions to the hierarchy of nonlinear LB

kinetic equations for stationary planar Couette flow at non-vanishing Knudsen numbers. By using a 16-speed two-dimensional lattice and kinetic boundary conditions, they have derived closed-form solutions for all higher-order moments and solved them analytically.

The results indicate that the LB hierarchy with larger velocity sets does indeed approximate kinetic theory beyond the NS level. If only for a simple setup, those exact solutions indicate that LB equipped with kinetic boundary conditions is able to carry quantitative nonhydrodynamic information, hence it can be regarded as a kinetic closure in its own right.

It thus appears that the extension of the LB formalism to higher-order lattices can result in an effective tool to probe deeper into nonequilibrium regimes beyond the hydrodynamic description (Gan *et al.*, 2015).

D. Fluctuating lattice Boltzmann method

For the case of nanoscale flows, reproducing thermal behavior implies that the LB method must incorporate the effects of statistical fluctuations. To achieve this goal, following in the footsteps of Landau-Lifshitz fluctuating hydrodynamics, Ladd (1993) added a source of random fluctuations of the momentum-flux tensor, namely,

$$\tilde{S}_i = \tilde{A} k_B T S_{ab}(c_{ia}c_{ib} - c_s^2 \delta_{ab}), \quad (33)$$

where $S_{ab} = (\partial_a J_b + \partial_b J_a)/2$ is the shear tensor and $\tilde{A}$ is the amplitude of the fluctuations.

The latter must be tuned so as to comply with the fluctuation-dissipation theorem. The resulting fluctuating NS equation reads

$$\rho\left(\frac{\partial}{\partial t}\vec{u} + \vec{u}\cdot\nabla\vec{u}\right) = \nabla\cdot(\overset{\leftrightarrow}{P} + \overset{\leftrightarrow}{S}) + \eta\nabla^2\vec{u} + \vec{G}, \quad (34)$$

where $\overset{\leftrightarrow}{P}$ is the fluctuation-free momentum-flux tensor and $\vec{G}$ is the body force acting on the fluid.

The LB scheme is modified accordingly by adding a stochastic source to the right-hand side:

$$\tilde{f}_i(\vec{r} + \vec{c}_i; t + 1) = (1 - \omega)\tilde{f}_i(\vec{r}; t) + \omega \tilde{f}_i^{\text{eq}}(\vec{r}; t) + \tilde{S}_i \quad (35)$$

as sketched in Fig. 10. The source term $\tilde{S}_i$ is local in space and time and acts at the level of the stress tensor and nonhydrodynamic modes, with no effect on mass and momentum conservation.

In actual practice, the source term $\tilde{S}_i$ is constructed via a set of the lattice eigenvectors $\{\chi_k\}$ with $k = 0$, $Q - 1$ orthonormal according to the scalar product $\sum_{m=0}^{Q-1} w_m \chi_{km}\chi_{lm} = \delta_{kl}$. In the D3Q19 scheme, the eigenvectors correspond to the kinetic moments: $k = 0$ is relative to density, $k = 1:3$ to the current, $k = 4:9$ to the momentum-flux tensor, the remaining $k = 10:(Q - 1)$ eigenvectors to nonhydrodynamic modes. The stochastic forcing reads as follows:

$$\tilde{S}_i = \sqrt{\frac{\rho k_B T \omega(2 - \omega)}{c_s^2}} \sum_{k=4}^{Q-1} w_i \chi_{ip} \mathcal{N}_k, \quad (36)$$
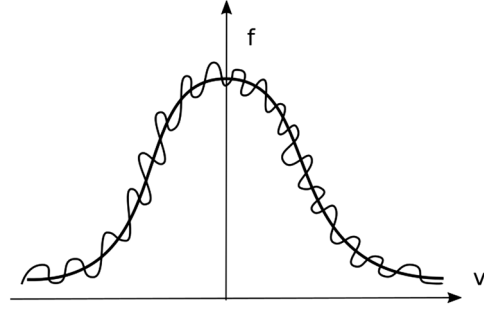
FIG. 10. A sketch of the fluctuating distribution $\tilde{f}$ (wiggly line), featuring rapid oscillations with respect to the nonfluctuating one f (smooth line). Fluctuating hydrodynamics emerges as a consequence of the stochastic component of the distribution function.

where $\mathcal{N}_k$ is a set of 15 random numbers with zero mean and unit variance. Given the fact that the thermal velocity is fixed and equals the underlying lattice speed c, the thermal mass is chosen in such a way as to obtain the thermal fluctuations according to $k_B T = m v_T^2$.

The stochastic forcing has been subsequently improved so as to produce consistent fluctuations at all spatial scales, in particular, at short distances where the effect on the translocating molecule is critical (Adhikari *et al.*, 2005; Dünweg, Schiller, and Ladd, 2007). The fluctuating LB method passes a number of litmus tests, particularly the compliance of velocity-velocity and force-force autocorrelation functions with the principle of stochastic kinetic theory, in particular, the fluctuation-dissipation theorem.

E. Lattice Boltzmann method for nonideal fluids

Boltzmann originally derived his equation under the assumption of diluteness, whereby molecular collisions take place as zero-ranged, instantaneous events, leading to large scattering angles. This rules out long-range, soft interactions giving rise to small-angle deflections. Such interactions, however, are of utmost importance for biological applications, since soft interlayers based on membranes and biopolymers define the spatial boundaries between different phases in biological systems. In aqueous media containing various ions, interactions are governed by a complex interplay of generic and specific interfacial interactions, typically controlled by dispersion and electrostatic forces, and locally shaped as hydrogen-bond networks, disulphide bridges, hydrophobic, entropy-induced interactions, etc.

Soft-core interactions can be included in the kinetic equation in the form of effective one-body forces, resulting from the collective interaction of a representative particle with the self-consistent environment (bath) due to all other particles. Formally, such an effective one-body force takes the following form:

$$\vec{F}_1 = \int \nabla_{r_1} V(\vec{r}_1 - \vec{r}_2) f_{12} d\vec{v}_2 d\vec{r}_2, \quad (37)$$

where $f_{12} \equiv f(\vec{r}_1, \vec{r}_2, \vec{v}_1, \vec{v}_2)d\vec{v}_2 d\vec{r}_2$ is the unknown two-body distribution and $V_{12} \equiv V(\vec{r}_1 - \vec{r}_2)$ is the two-body atomistic

potential. Following a customary practice, one writes $f_{12} = f_1 f_2 g_{12}$, where g_{12}, the two-body correlation function, collects the two-body physics. The exact form of the correlation function is known exactly only in a few cases; yet one can introduce several useful *Ansätze* which turn the formal expression (37) into an operational one. A typical *Ansatz*, which has proven fruitful for LB modeling of nonideal fluids, looks like

$$\vec{F}(\vec{r}_1) = \psi(\vec{r}_1) \int G(\vec{r}_1, \vec{r}_2) \psi(\vec{r}_2)(\vec{r}_2 - \vec{r}_1) d\vec{r}_2, \quad (38)$$

where $G(\vec{r}_1, \vec{r}_2)$ is a model Green's function and $\psi[\vec{r}] \equiv \psi[\rho(\vec{r})]$ is a local functional of the fluid density $\rho(\vec{r}_1, t)$. Equation (38) is quite general and allows a wide degree of latitude in *modeling* nonideal fluid interactions.

F. Pseudopotential models

The most popular lattice transcription, due to Shan and Chen (1993, 1994), reads as follows:

$$\vec{F}(\vec{r}) = \psi(\vec{r}) \sum_i w_i G_i \psi(\vec{r} + \vec{c}_i) \vec{c}_i. \quad (39)$$

The sum runs over the prescribed set of neighbors, typically the first Brillouin region in the original version, and subsequently extends to the second or even the third one. Typically, all discrete speeds in the same Brillouin region share the same G_i, so that in the original Shan-Chen (SC) formulation, there is just one coupling strength G; see Fig. 11. This is nonetheless sufficient to generate the main ingredients of nonideal fluids, namely, a nonideal equation of state supporting phase transitions, as well as surface tension. Moreover, this limitation can be readily lifted by extending the Shan-Chen interaction beyond the first Brillouin region, thereby explicitly accounting for both repulsive and attractive interactions.

The Shan-Chen expression delivers a nonideal equation of state of the form

$$p = \rho c_s^2 + \frac{G}{2} c_s^2 \psi^2[\rho]. \quad (40)$$

By choosing the generalized density in the form

$$\psi[\rho] = 1 - e^{-\rho}$$

it is readily checked that the Shan-Chen fluid becomes critical for $G < G_{\text{crit}} = -4$ at a critical density $\rho_c = \ln 2$. Note that even though the interaction is purely attractive ($G < 0$), no pileup instabilities take place because the force becomes increasingly faint as the density increases. This is the reason for using the generalized density $\psi[\rho]$ instead of the physical one ρ. This is an expedient trick to trigger phase transitions without any repulsive force. The translation from the force to the LB source S_i proceeds through a systematic expansion in lattice Hermite polynomials. To leading order $S_i \sim \vec{F} \cdot \vec{c}_i$, but higher-order terms, at least quadrupole ones, are definitely needed to obtain accurate results.
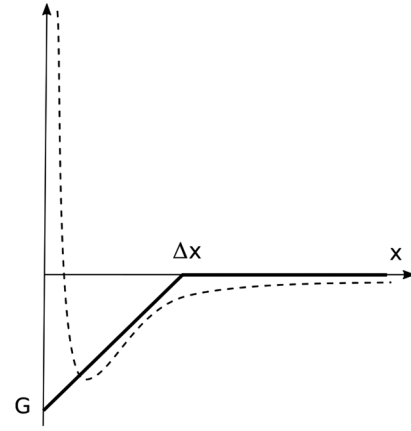

FIG. 11. Sketch of the Shan-Chen piecewise linear potential extending over the first Brillouin lattice cell at a distance Δx. The potential mimics the attractive tail of a van der Waals potential (dashed line) while the repulsive one is quenched to zero. Notwithstanding the absence of hard-core repulsion, the Shan-Chen potential does not cause any unstable density pileup because, in the high-density limit $\rho \gg 1$, the generalized density $\psi(\rho) = 1 - e^{-\rho}$ flattops to a constant 1, yielding a zero gradient, hence zero force.

More generally, thick interfaces lead to large values of the Cahn number, namely, the ratio of the interface width to a typical mesoscopic scale, say the droplet (bubble) diameter:

$$\text{Cn} = \frac{w}{D}.$$

For most applications D is in the range of tens, up to 100, microns, leading to very small Cahn numbers $\text{Cn} \sim 10^{-5}$. Replicating this number in LB simulations is totally unviable, for it would amount to placing $O(10^5)$ lattice spacings across the particle diameter. LB simulations are forced to operate at much higher Cahn numbers of the order of 0.01–0.1, which means that potential inaccuracies due to finite Cahn number effects need to be carefully monitored.

Lack of sufficient symmetry also leads to the appearance of spurious currents, which may seriously affect the physical results whenever the density ratio between the light and dense phases is above 30–50, a problem which is further exacerbated in the case of fast-moving interfaces. Finally, the model is not thermodynamically consistent, as it is not derived from a mean-field free-energy functional. That limitation is, however, less serious than it seems, since the method is, actually, equipped with a quasifree energy (Sbragaglia *et al.*, 2009). As a matter of fact, more disturbing, instead, is the fact that the Shan-Chen state equation features a sound speed smaller in the liquid with respect to the vapor phase, with significant consequences on the stability of the light phase across the interface.

Most of these limitations have been significantly mitigated, if perhaps not fully resolved, by subsequent developments. Among others, particularly noteworthy for the simulation of soft-glassy materials, is the so-called multirange pseudopotential method.

The main idea is to augment the original Shan-Chen short-range attraction in the first Brillouin region (belt, in LB jargon), say the D3Q27 lattice in three dimensions, with a repulsive interaction acting on the second Brillouin region, i.e., including discrete velocities up to $\sqrt{5}$. The main advantage of this formulation is that a proper tuning between the first-belt attraction and the second-belt repulsion makes it possible to achieve smaller values of the surface tension, thereby permitting one to sustain long-lived, multidroplet configurations with a highly complex interfacial dynamics. Despite its many limitations, the SC method remains the most popular version of the LB method for nonideal fluids, mostly on account of its simplicity and transparency.

However, for applications leading to complex interfacial dynamics more advanced schemes are required.

1. Free-energy models

Another successful route to LB schemes for nonideal fluids is lattice density functional theory, whereby the nonideal physics is encoded within a free-energy functional of the form

$$\mathcal{F}[\rho] = \int \phi(\rho, \nabla\rho) d\vec{r}, \tag{41}$$

where ϕ is a local functional of the fluid density and its gradients (Swift *et al.*, 1996). The former encodes the nonideal equation of state, while the latter describes the effect of surface tension.

Variational minimization over density changes delivers the equations of motion of the nonideal fluids, governed by the Korteweg pressure tensor:

$$P_{ab}(\vec{r}) = p(\vec{r})\delta_{ab} + \kappa \partial_a \rho \partial_b \rho, \tag{42}$$

where

$$p(\vec{r}) = p_0(\rho) - \kappa[\rho\Delta\rho - \tfrac{1}{2}(\partial_a\rho)^2]$$

is the nonlocal pressure, consisting of the bulk contribution $p_0(\rho)$, fixing the equation of state, plus an interface contribution associated with surface tension. In Eq. (42), ∂_a stands for the space derivative along direction $a = x, y, z$ and κ is a tunable coefficient controlling the surface tension.

From a practical standpoint, the free-energy formulation leads to nonlocal equilibria, which involve second-order derivatives of the density field, thus taxing the simplicity of the method, and sometimes its stability as well, as compared to the pseudopotential method. Nevertheless, the free-energy method has found broad use for many applications such as microflows over geometrically or chemically patterned surfaces and various types of droplet motions (Liu *et al.*, 2014).

Several variants have been developed over the years, which have considerably improved over the original versions, especially in terms of reaching higher density ratios without compromising numerical stability, among others, high-order finite-difference schemes (Lee and Lin, 2005) and mergers with flux-limiting methods (Sofonea *et al.*, 2004). These methods are currently used to simulate a variety of multiphase and multicomponent fluid applications, such as

Rayleigh-Taylor instabilities, droplet collisions, cavitation, and free-surface flows. For a review, see the recent books by Montessori and Falcucci (2018) and Succi (2018) and references therein.

2. Chromodynamic models

Another class of LB methods for nonideal fluids which has been revamped in the recent past is the so-called color-gradient model. Here the two phases or components, call them red and blue for convenience, segregate based on a top-down "coloring" rule, which sends the particles along the color gradient, namely, red toward red and blue toward blue, thereby triggering an instability leading to interface formation (Gunstensen *et al.*, 1991).

The recoloring amounts to correcting the red and blue populations as follows:

$$f_i^{R,B} = \phi^{R,B} f_i^* \pm \beta \phi^R \phi^B \mu_i f_i^{\text{eq},0}, \tag{43}$$

where $\phi^{R,B} = \rho^{R,B}/(\rho^R + \rho^B)$ is the mass density fraction, and μ_i is the cosine of the angle between the color gradient $\vec{G} = \nabla(\rho_B - \rho_R)$ and the ith discrete velocity. In Eq. (43) $f_i^* = f_i^{R,*} + f_i^{B,*}$, where the asterisk indicates the population after the application of the nonideal force stemming from surface tension, and $f_i^{(\text{eq},0)} = f_i^{(\text{eq},0),R} + f_i^{(\text{eq},0),B}$, where superscript $(\text{eq}, 0)$ denotes the equilibria at zero flow. Finally, β is a free parameter controlling the strength of the recoloring procedure, to all effects and purposes an antidiffusive operator. The crucial term is the second one on the right-hand side of Eq. (43), which, by construction, is active only at the interface between the two components. For more details, see Leclaire *et al.* (2017) and Montessori, Lauricella, and Succi (2018).

The interface is then stabilized by means of a chromodynamic force proportional to the amplitude of the color gradient, but opposite to it, so as to level out the deficit of one species over the other (color gradient). Although essentially rule driven, modern variants of this scheme have proved capable of accessing parameter regimes which appear to be off limits for Shan-Chen schemes and extensions thereof, as well as for free-energy methods. For instance, such methods have been recently applied to the design of microfluidic devices for droplet generation, such as flow focusers and step emulsifiers (Montessori, Lauricella, La Rocca *et al.*, 2018; Montessori, Lauricella, Succi *et al.*, 2018).

3. Entropic models for multiphase flows

The lattice Boltzmann method can also be formulated by minimizing a suitable lattice H function of the form (Kullback-Leibler entropy)

$$H[f] = \sum_i f_i \log(f_i/w_i),$$

where w_i are suitable lattice weights. The resulting scheme takes the usual form of a standard LB method, with the crucial twist that the relaxation time is adaptively adjusted in such a

way as to secure compliance with the second principle of thermodyamics, namely,

$$\frac{dS}{dt} = \int H[f] d\vec{r} d\vec{v} \geq 0.$$

Leaving a detailed description to the original literature, here we just mention that compliance with the second principle translates into a significant enhancement of numerical stability (Karlin *et al.*, 1998). For this reason, the ELB method has found profitable use for the simulation of low-viscous flows typical of fluid turbulence.

More recently, the ELB method was extended to the case of multiphase flows and shown to provide stability benefits also in the viscous regime of relevance to many biological applications (Montessori *et al.*, 2017; Wöhrwag *et al.*, 2018). Although these developments are too recent to permit a solid statement, the results appear encouraging and leave hope that the entropic version of the LB method may become a major player in the field for the years to come.

V. FLUIDS AND PARTICLES: THE LATTICE BOLTZMANN–PARTICLE DYNAMICS SCHEME

The multiphase LB schemes discussed in the previous section have generated a mainstream of applications in soft-matter research, since they permit one to deal with flows of major dynamic and morphological complexity, such as foams and emulsions. However, they are unsuited to handle rigid bodies suspended in the continuum phase, nor can they describe in detail mechanical properties of deformable objects, say membranes, vesicles, cells, and other biological bodies. To this purpose, the LB method needs to be explicitly coupled to particle methods, tracking the dynamics of the biological bodies immersed in the flow.

Indeed, most flows of biological interest consist of biological bodies of assorted nature: cells, polymers, proteins, floating in a fluid solvent, typically water. Such flows often operate at low, often near-zero, Reynolds number (Purcell, 1977), but this does not mean that hydrodynamic interactions (HI) can be neglected. To the contrary, HI have repeatedly been shown to accelerate a variety of nanoscale biological transport processes, such as biopolymer translocation across nanopores, amyloid aggregation of proteins in the cell, and other related phenomena. This explains why the combination of the LB method with particle dynamics has made the object of extensive research and the development of multiple simulation schemes.

A. The extended particle model (EPM)

Flows with suspended objects have been modeled since the early days of LB research, starting with the trailblazing work of Ladd 1994a, 1994b). Ladd's original method consists of tracking the motion of rigid spherical bodies under the impact of the surrounding solvent hitting the surface of the body. The scales pertaining to body and solvent are fairly separate, since the former is much larger and heavier than the latter; hence the mass ratio m/M serves as a suitable scale separator.

In the EPM, the exchange of momentum between particles and LB fluid is a boundary-collision method whereby the suspended bodies interact with each other only through the mediation of local collisions with the solvent molecules (LB). The solvent-body collision is conservative, and the momentum lost (gained) by the solvent to the body is gained (lost) by the body on the solvent. The method therefore is based on the local exchange of momentum by computing the amount of momentum that every population hitting the body surface exchanges with the latter. Being the total force $\vec{F}$ and torque $\vec{T}$ the sum of fluid-body momentum exchange (or drag interactions), direct particle-particle mechanical interactions or, in case of microscopic conditions, stochastic forces due to the Brownian motion, the body linear momentum $\vec{p}$ and angular momentum $\vec{l}$ obey the classical equations of motion

$$\frac{d\vec{p}_i}{dt} = \vec{F}_i, \quad \frac{d\vec{l}_i}{dt} = \vec{T}_i, \quad i = 1, N_p. \tag{44}$$

It is worth mentioning that the fluid-particle coupling is hydrokinetic in nature because it is treated at a collisional level rather than at a hydrodynamic level, so that hydrodynamic forces such as long-time tails naturally emerge at times larger than the LB marching one. The approach naturally takes into account the solvent fluctuations, if present, since the latter are transmitted across the body surface analogously to the drag forces; see Fig. 12.

Since the solvent-body collisions are conservative, no extra stochastic source is needed on the body side.

Ladd's method finds its best use in simulating colloidal suspensions of rigid particles and in this sense has found a comparatively limited application in the biological context. However, it provides the first and reliable example of coupling between the LB method and suspended bodies. At the same time, the method shows some inaccuracy due to the lattice discreteness, especially for near-contact colloid-colloid interactions, occurring below the grid scale where the LB method cannot resolve the lubrication forces. Some variants have been developed in the literature, making use of grid refinement and/or dynamic interpolations. Not surprisingly, they add only to the computational complexity of the method, which is comparatively laborious on its own, as it demands full knowledge of the local fluid-solid connectivity, namely, the dynamic list of fluid nodes interacting with the solid ones.

B. The point-particle model (PPM)

A qualitatively different strategy was proposed by Ahlrichs and Dünweg (1998). This is a minimal way to embed pointlike particles in the LB fluid which, as opposed to Ladd's method, is entirely *force driven*, hence metric instead of topologic. In essence, particles carry a phenomenological friction coefficient γ_T (also known as the bare friction coefficient) and the fluid exerts a drag force based on the instantaneous difference between the particle and fluid velocities, reading

$$\vec{F}^{\text{drag}} = -\gamma_T [\vec{V} - \vec{u} \delta(\vec{r} - \vec{R})]. \tag{45}$$
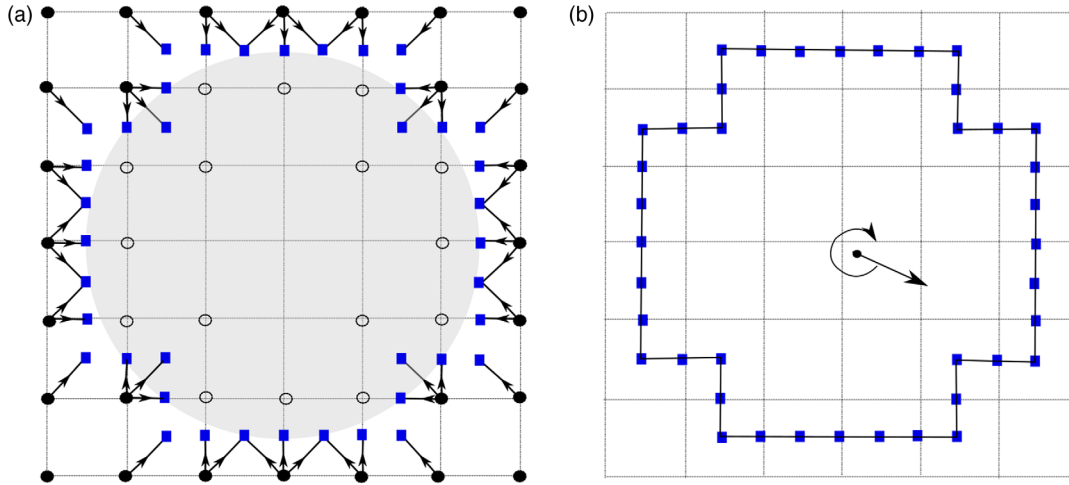
FIG. 12. Ladd's method for a suspended (spherical) body: (a) the body-fluid interface is tracked by searching the mesh points inside and outside the particle (Ladd, 1994a). Here a modified bounceback scheme is applied to account for the local exchange of momentum. The midpoint of the connecting links identifies the staircased surface that corresponds to the no-slip condition. (b) the particle center moves in the continuum according to the momentum exchange, and the rotational motion is accounted for by computing the total torque.

The addition of the point force into the fluid equations introduces a singularity into the flow field, as represented by Dirac's delta function. This is automatically smoothed out by interpolation procedures, but requires nonetheless due care in the numerical treatment.

On the other hand, the flow field around a finite-sized particle is generated by a distributed force located around the particle position, as shown in Fig. 13.

In the scheme, the flow field is treated as everywhere finite and the force density acts onto the LB fluid at position **r** as

$$\vec{G} = -\vec{F}^{\rm drag} I(\vec{r}, \vec{R}), \qquad (46)$$

where $I(\vec{r}, \vec{R})$ is a function interpolating between the particle position $\vec{R}$ and the surrounding mesh node $\vec{r}$. In its original formulation, the method replaced $\vec{u}\delta(\vec{r} - \vec{R})$ and $I(\vec{r}, \vec{R})$ by the values of these fields at the mesh node nearest to the particle position.

A refined version resorts to a simple linear interpolation scheme using the mesh points on the elementary lattice cell containing the particle. Denoting the relative position of the particle in this cell by (d_x, d_y, d_z) with the origin being at the lower left front edge, one defines $\delta_{0,0,0} = (1 - d_x)(1 - d_y)(1 - d_z)$, $\delta_{1,0,0} = d_x(1 - d_y)(1 - d_z)$ etc., as the interpolation weights. The interpolated velocity then reads $\vec{u}\delta(\vec{r} - \vec{R}) \simeq \sum_{r \in {\rm cell}} \delta_r \vec{u}(\vec{r})$, where the sum is over the mesh points on the considered elementary lattice cell. The force exerted back to the fluid can then be chosen with the same weight coefficients or by spreading the force equally on the edges of the cell, the corresponding rule to preserve linear momentum being easily derived.

Because of the regularization of the point force, one should expect that the mobility of the suspended particle is not simply given by the bare friction coefficient γ_T, but is somehow renormalized. Indeed, the mobility is given by the sum of the bare mobility and a Stokes-type contribution due to the lattice discretization. The effective friction coefficient

relates to the bare one via $1/\gamma_T^{\rm eff} = 1/\gamma + 1/g\eta\Delta x$, where $g \simeq 25$ is a geometrical correction coefficient (Ahlrichs and Dünweg, 1998).

The PPM finds application in simulating microscopic systems with a stochastic term added on each particle in
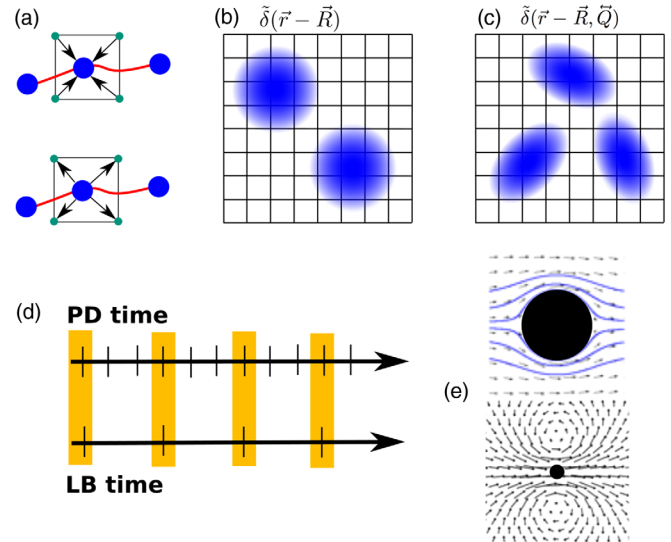
FIG. 13. Examples of immersed bodies in the LBPD approach. (a)–(c) The different coupling methods between particles and fluid as detailed in the text: (a) The PPM with pointlike particles exchanging information with the nearest grid node. (b) The DPM for diffused isotropic particles. (c) The DPM for diffused aniso-tropic particles. (d) The time-stepping method between particle dynamics (PD) and fluid dynamics (LB). The time-stepping scheme of the LB and PD individual components can be either synchronous or asynchronous. Typically, the PD component ticks more frequently than the LB component, allowing for more efficient simulations. (e) A typical configuration of the fluid around a moving particle. Hydrodynamic response manifests itself as Stokes flow field (upper part) and long-range flow structure around a spherical particle or its far-field flow pattern (lower part).

addition to the fluctuating LB bath. In this way, the particle does not leak energy on average and the fluctuation-dissipation balance maintains a well-defined temperature of the fluid-particle system. Again, to balance and preserve momentum, such stochastic force is also restituted to the fluid with an opposite sign. The presence of long-time tails, that is, the inherent hydrodynamically sustained motion of the moving particle generating a long-time decay of velocity, has also been observed (Ahlrichs and Dünweg, 1998). On the downside, the PPM is permeable to fluid momentum as the local force is unable to create enough resistance to the incoming flow, and the classical Stokes-like picture of the streamlines does not apply.

Owing to its inherent simplicity, the PPM was first recognized to be useful to simulate topologically connected particles, such as polymers, in the presence of hydrodynamic interactions. Whenever the embedded particles represent microscopic or mesoscopic objects, such as atoms or the beads of a polymer, mass diffusion plays a role comparable to mechanical and hydrodynamic forces. Another interesting extension involves coupling PPM with the Shan-Chen multi-component LB to simulate efficiently complex fluid-fluid interfaces. The idea is to introduce a solvation free energy for the particle-fluid interaction proportional to the fluid density gradients (Sega *et al.*, 2013) and reads as follows:

$$\vec{F}^{\text{solv}} = -\zeta \nabla \rho. \tag{47}$$

Such force drives particles toward maximal or minimal density gradients, depending on the sign of the coupling coefficient ζ. The approach is particularly appealing when used in conjunction with multiphase conditions. As a matter of fact, the approach makes it possible to treat different multiphase fluids in the presence of suspended molecules, such as amphiphilic molecules as surfactants or by adding another

level of detail with electrostatics, polyelectrolytes in bicomponent fluids.

C. The diffused-particle model (DPM)

An extension of the PPM represents particles with finite-size extension, still relying on a force-based mechanism. Given the finite extension, it is well suited for anisotropic particles, providing a handy computational flexibility for the description of biological suspensions, cellular compartments, or even entire cells (recalling that the interior of the cell is anisotropic due to intracellular organelles). The strategy is to consider the particle rototranslational response as originating from the coupling of the finite-size extension of the particle with the fluid momentum and vorticity. The particle is an effective diffused body, with no need to track its boundary to control its coupling with the environment.

The particle shape is described as an ellipsoid having three major radii along the three principal directions ξ_α with $\alpha = 1$, 2, 3. To fit in the discrete nature of the lattice, ξ_α are taken as three integers, one for each Cartesian component (such requirement can be lifted but a few interesting properties described later would be lost). The rototranslation is governed by rigid body dynamics of Eqs. (44). The particle rotational state is encoded by the matrix $\overset{\leftrightarrow}{Q}$ whose rows are three orthogonal unit vectors aligned along the principle axis of the particle, that is, the basis to transform between the laboratory and the moving frames. The rototranslational state is given by the tensorial product

$$\tilde{\delta}(\vec{r}, \overset{\leftrightarrow}{Q}) = \prod_{\alpha = x,y,z} \tilde{\delta}[(Q\vec{r})_\alpha], \tag{48}$$

where

$$\tilde{\delta}(y_\alpha) = \begin{cases} \frac{1}{8}\left(5 - 4|y_\alpha|/\xi_\alpha - \sqrt{1 + 8|y_\alpha|/\xi_\alpha - 16y_\alpha^2/\xi_\alpha^2}\right) & |y_\alpha/\xi_\alpha| \leq 0.5, \\ \frac{1}{8}\left(3 - 4|y_\alpha|/\xi_\alpha - \sqrt{-7 + 24|y_\alpha|/\xi_\alpha - 16y_\alpha^2/\xi_\alpha^2}\right) & 0.5 < |y_\alpha/\xi_\alpha| \leq 1, \\ 0 & |y_\alpha/\xi_\alpha| > 1 \end{cases}$$

is a shape function having compact support and the normalization property $\sum_r \tilde{\delta}(\vec{r}) = 1$. Typically $\xi_x = \xi_y = \xi_z = 2$ corresponds to a spherically symmetric diffused particle with a support extending over 64 mesh points. The translational response of the suspended body is designed according to the fluid-particle exchange kernel

$$\vec{\phi}(\vec{r}, \vec{R}, \vec{V}) = -\gamma_T \tilde{\delta}(\vec{V} - \vec{u}),$$

where $\tilde{\delta} \equiv \tilde{\delta}(\vec{r} - \vec{R}, \overset{\leftrightarrow}{Q})$ and the hydrodynamic DPM force is obtained via integration over the particle spatial extension. It reads as follows:

$$\vec{F}^{\text{drag}} = \sum_r \vec{\phi}(\vec{r}, \vec{R}, \vec{V}) = -\gamma_T(\vec{V} - \tilde{\vec{u}}),$$

where $\tilde{\vec{u}} \equiv \sum_r \tilde{\delta}\vec{u}$.

The coupling between the body motion and the fluid vorticity is represented by a rotational kernel

$$\vec{\tau}(\vec{r}, \vec{R}, \boldsymbol{\omega}) = -\gamma_R \tilde{\delta}(\vec{\Omega} - \vec{\omega})$$

with γ_R a rotational coupling coefficient. The corresponding drag torque is

$$\vec{T}^{\text{drag}} = \sum_r \vec{\tau}(\vec{r}, \vec{R}, \vec{\omega}) = -\gamma_R(\vec{\Omega} - \tilde{\vec{\omega}}),$$

where $\tilde{\vec{\omega}} \equiv \sum_r \tilde{\delta}\vec{\omega}$.

In the DPM, the smooth function $\tilde{\delta}$ is a very natural interpolating device, and moving information back and forth between the grid and the particles bears a significant influence

on the effective hydrodynamic size of the body. In particular, spherical particles acquire an effective radius given by

$$\frac{R_{\text{eff}}}{R} = 1 + a\frac{\Delta x}{R} - b\frac{R}{L}.$$

In this equation, a and b are positive numerical prefactors taking into account the range of the interpolators and finite-size effects. The former make the effective radius larger, hence larger dissipation, due to defective autocorrelations triggered by grid discreteness. The latter have the opposite sign because the lack of large-scale modes above the box size results in lesser dissipation.

These corrections are crucial to the purpose of matching the diffusion coefficient of the simulated bodies to the physical one (Ahlrichs and Dünweg, 1998; Dünweg, Schiller, and Ladd, 2007).

D. Immersed boundary methods

The hydrodynamic force and torque acting on the DPM are obtained via integration over the particle spatial extension (volume force). A different approach represents the finite extension of particles by tracking only the particle boundary degrees of freedom, the so-called immersed boundary method (IBM), developed by Peskin (2002) decades ago, to deal with immersed moving boundaries within fluid flows. Borrowing from Ladd's approach, on the one hand, and Ahlrichs and Dünweg, on the other hand, the LB-IBM procedure is both boundary driven and force driven. The major benefit is that the body surface is treated like a deformable membrane, emanating an elastic force field toward the outside fluid. In the following:

$$\vec{F}_f(\vec{r};t) = \int_{\mathcal{M}} \vec{F}_m(\vec{R},t)\delta[\vec{r} - \vec{R}(t)]d\vec{R}, \qquad (49)$$

where $\vec{r}$ is the generic fluid location, $\vec{R}$ runs over the two-dimensional membrane, and $\vec{F}_m$ is the force acting on the membrane at location $\vec{R}$. The latter is usually computed as the divergence of the elastic stress tensor of the membrane. The membrane equation of motion is given by the Lagrangian condition

$$\frac{d\vec{R}}{dt} = \vec{u}_f(\vec{R},t), \qquad (50)$$

where $\vec{u}_f$ is the fluid velocity extrapolated to the membrane location, i.e.,

$$\vec{u}_f(\vec{R},t) = \int_{\mathcal{M}} \delta(\vec{r} - \vec{R})\vec{u}(\vec{r})d\vec{r}. \qquad (51)$$

The fluid equation of motion is the standard (Eulerian) LB, with the force given by the numerical version of the integral in Eq. (49).

Note that the accuracy and efficiency of the LB-IBM scheme are highly dependent on the discrete representation of the Dirac delta. This is usually replaced by piecewise polynomials extending over a few lattice sites, typically four

for cubic splines, sometimes also known as "smoothed particles." The LB-IBM is a fully coupled nonlinear and nonlocal Eulerian-Lagrangian scheme; hence it presents a demanding computational task. However, it provides the major benefit of smoothness, due to the integral nature of the convolutions which control the exchange of information between the LB fluid and the IBM membrane. The LB-IBM scheme is gaining increasing popularity for soft-matter applications involving the interaction of micronanoscale fluids with deformable suspended bodies.

E. Chemical specificity and coarse graining

Realistic microscopic biological simulations set a key quest for chemical specificity. As summarized in the announcement of the 2013 Nobel prize in Chemistry (http://www.nobelprize.org/nobel_prizes/chemistry/laureates/2013/), a tremendous amount of information has been gained during the last 40 years by using molecular dynamics to simulate proteins, that is, by reproducing molecular forces to high accuracy and the complex motion of biological settings based on Newton's equations of motion.

The hallmark of protein simulations has been the discovery that complex macromolecules can be represented by force fields that include intra-aminoacidic and interaminoacidic interactions via comparatively compact potential functions. Several force fields, such as the well-known Charmm and Amber force fields (Ponder and Case, 2003; Guvench and MacKerell, 2008; Lindorff-Larsen et al., 2012), have been developed in the all-atom context, that is, by taking into consideration all possible atoms stemming from the macromolecule and the aqueous solvent. The all-atom strategy was dictated by the high level of heterogeneity of biological interactions, ranging from the local hydrogen bonds, to dispersion forces, to the long-range electrostatic interactions.

Computing such diverse forces at high-performance rates mandates sophisticated parallel algorithms and dedicated hardware. In order to unveil macromolecular dynamics and, most notably, folding within the characteristic *funneled* free-energy landscapes (Levitt and Warshel, 1975; Frauenfelder, Sligar, and Wolynes, 1991; Bryngelson et al., 1995), to large stretches of time, up to milliseconds, few outstanding simulations have been carried out to date, by exploiting custom or specialized hardware (Shaw et al., 2008; Feig et al., 2017). While providing extraordinary results, such efforts are still isolated and, more importantly, cannot deal with large size systems, due to the large memory requirements required in practice, and the large hardware and power costs of extreme-scale simulations.

For these reasons, more simplified versions of force fields have emerged in recent years, the so-called coarse-grained force field (CGFF) with the aim to tackle large systems while retaining the desired degree of chemical specificity and accuracy in place (Kamerlin et al., 2011). A few examples of CGFF are currently available (MARTINI, OPEP) (Chiricotto, Melchionna et al., 2016; Chiricotto et al., 2017) and they are being deployed to study many different macromolecular systems, ranging from polypeptides to polynucleotides. A key point of the CGFF approach is the mesoscale nature of the molecular representation, in fact, multiple atoms are replaced by effective particles.

Such particles are then connected by bonding potentials that enforce local backbone connectivity and structure. Other non-bonding interactions are used to enforce dispersion and hydrogen bond forces packaged into a lumped form.

Importantly, the CGFF was developed to remove the explicit need to represent the molecular details of the solvent and as such, solvent-mediated interactions, as much as electrostatics, are effectively recast in terms of the CGFF potential form. It is true, though, that coming up with a consistent form of the solvent-mediated interactions is not an easy task and some force fields, such as the MARTINI one, use some form of particle-based method to represent a minimalistic type of solvent. Setting aside details, the CGFF provides a major step toward biosimulations. In future years, we are most likely to witness more force fields, with the goal of dispensing with the explicit representation of the solvent.

The most relevant aspect in the present context is that the CGFF approach sits well with the mesoscale description for the embedding solvent and the LB method provides an ideal partner to simulate large protein assemblies. Since thermal fluctuations are key at this scale, the simulations require Langevin terms as well as thermal fluctuations on the LB side; see Fig. 14.

As the basic form of the LB method contains only ideal thermodynamics, the CGFF does not need to be modified with respect to the original version used in the absence of HI. This holds true because from the statistical mechanics viewpoint, effective forces under equilibrium conditions do not depend on hydrodynamic, velocity-dependent interactions (Hansen and McDonald, 1990; Noid *et al.*, 2008). Conversely, under nonequilibrium conditions, say subject to external

macroscopic flow, the CGFF requires changes. As the need arises in the direction of reintroducing solvent-mediated interactions stemming from nonideal thermodynamics or hydrogen-bond interactions, one may think of reformulating the CGFF to account for an explicit nonideal thermodynamics of the solvent. Many research works and applications ahead are awaiting for full exploration.

As for any simulation of proteins in solution, a word of caution is in order about the usage of mesoscale simulations to describe the aqueous solvent as a continuum. The LB mesh spacing Δx is defined as a coarse-grained representation of the collective kinetic behavior of a group of solvent molecules. In order to observe hydrodynamic behavior down to the mesh spacing distances, the mean-free path is usually expected not to exceed Δx. In the liquid state, the molecular mean-free path extends over just a few angstroms, thus commanding sub-nanometric lattice spacings to allow the emergence of hydrodynamic behavior at scales larger than Δx.

As shown by Horbach and Succi (2006), this strategy is effective for simulating nanofluidic coherent patterns in close agreement with those obtained by molecular dynamics simulations.

1. Life at low Mach and Reynolds numbers: Numerical caveats

In this section we briefly touch on some caveats that need to be taken into account in the LBPD simulations at vanishingly small values of the Reynolds and Mach numbers.

The three main dimensionless groups characterizing incompressible flows are the Reynolds number $\mathrm{Re} = UL/\nu$, the Mach number $\mathrm{Ma} = U/C_s$, and the Knudsen number $\mathrm{Kn} = \lambda/L$, where U is the flow speed, C_s is the physical sound speed, L is a typical length scale, and λ is the molecular mean-free path. Elementary manipulations show that the three groups are related through the so-called von Karman relation:

$$\mathrm{Re}\,\mathrm{Kn} = \mathrm{Ma}.$$

In order for an ensemble of molecules to behave collectively as a fluid, the Knudsen number must be significantly below unity, $\mathrm{Kn} \ll 1$, which means that low Reynolds numbers imply even smaller Mach numbers. Since the LB method explicitly tracks sound waves, this implies extremely small time steps. A few numbers help in clarifying the point. Consider a nanoscale LB simulation of water with a lattice spacing $\Delta x = 10^{-9}$ m. Given that the kinematic viscosity of water is $\nu = 10^{-6}$ m^2/s, using $\nu_{\mathrm{LB}} = 0.1$ implies a time step $\Delta t = (\nu/\nu_{\mathrm{LB}})\Delta x^2 = 10^{-13}$ s. This shows that the LB method ticks "just" 2 orders of magnitude above molecular dynamics. While appropriate for nanoscale simulations, it is clear that such small time steps do not allow one to cover macroscopic time spans. At a given spatial resolution, the only way to increase the time step is to increase the lattice viscosity ν_{LB}, but, as explained earlier, taking ν_{LB} above unity runs against the hydrodynamic constraint $\mathrm{Kn} < 1$. Hence LB simulations in the Stokes limit $\mathrm{Re} \to 0$ are restricted to very small time steps. One way of short-circuiting the problem is to artificially enhance the Mach, hence the Reynolds number, in the hope that the physics in point be rather insensitive to the specific value of these numbers, as long as they are both well below

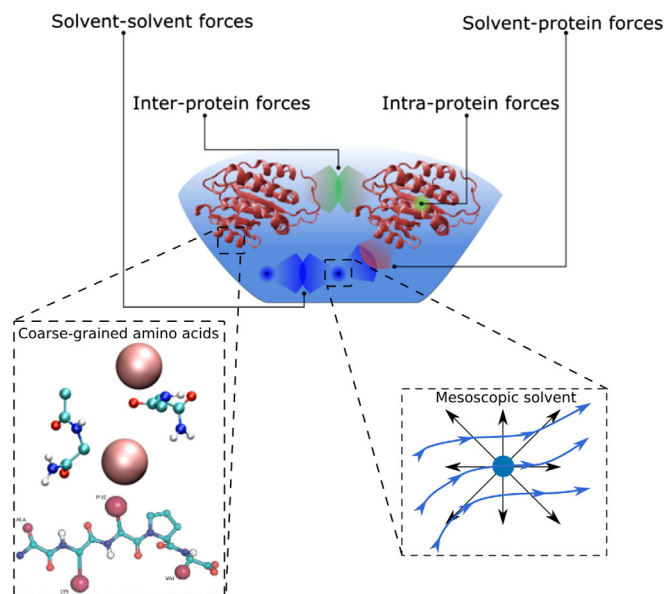FIG. 14. Proteins immersed in a LB solvent are treated as mesoscopic objects. Consequently, their representation is coarse grained, that is, a level of detail based on a handful of pseudoatoms per amino acid (five pseudoparticles in the current case from the OPEP model) (Chebaro, Pasquali, and Derreumaux, 2012). The pseudoparticles are needed to encode the details of the peptidic backbone and effective interparticle interactions.

unity. Under such benevolent conditions, the Reynolds number, say 10^{-6}, could be safely inflated to, say Re $= 10^{-2}$, without incurring any major disruption of the physics at hand. Needless to say, this "inflationary stratagem" works only as long as the physics exhibits analytic and nonsingular dependencies on the Reynolds number. Although characterized by the onset of nonlocal, long-range interactions, the limit Re $\to 0$ is, in principle, a nonsingular one, thus leaving some chances to the inflationary strategy previously discussed.

F. LB versus PD resolution

The LBPD scheme relies on two essential time scales for both the fluid solvent dynamics, controlled by kinematic viscosity ν, and the molecular dynamics, controlled by the friction, or drag coefficient γ_T. The two are related via Stokes law:

$$\gamma_T = 3\pi\eta\mathcal{D}/M,$$

where $\eta = \rho\nu$ is the solvent dynamic viscosity, $\mathcal{D}$ is the particle (protein) equivalent diameter, and M its mass. In the LBPD scheme, however, fluid and particles are handled as numerically independent quantities, so as to achieve the desired level of coarse graining between fluid and particle degrees of freedom. By obeying the fluctuation-dissipation principle $D = kT/\gamma_T$, at a given level of thermal fluctuations, mass diffusivity and coupling parameter are changed as a single parameter. Another important quantity is the ratio of solvent viscous and solute mass diffusivities, the so-called Schmidt number,

$$\text{Sc} = \frac{\nu}{D}.$$

In an ideal gas Sc $= 1$, but in a liquid it typically exceeds 100.

When simulating proteins within the LBPD scheme, a conservative approach is to represent proteins at a coarse-grained level by taking a mesh spacing $\Delta x = 2$ Å and a time step of $\Delta t = 1$ fs. The latter guarantees that the protein internal motion is duly resolved by the PD solver, whereas the mesh spacing guarantees that the local hydrodynamic signal is captured by the LB solver. Since proteins and biological aggregates range in size from tens to thousands nm in diameter, such mesh spacing fully resolves the mesoscale structure. One may reasonably wonder whether at the subnanometric scales the notion of Boltzmann distribution functions still makes sense, due to the dominance of statistical fluctuations. This is indeed possible by introducing the notion of effective thermal mass to be detailed in the sequel.

The starting setting is such that a protein particle sits typically inside a mesh voxel, a comfortable situation for numerical production purposes. On the other hand, it is often desirable to increase Δx to minimize the computational burden. Under such conditions, the numerical effort involved in handling the LB solver dominates the MD component, typically by a factor of 3–5. However, it is well known that, in principle, resolving the subnanometric hydrodynamics is unnecessary, since hydrodynamic interactions typically set in above the 3–5 Å scale.

Under operating conditions, it is convenient to fix the time step Δt, as this is a rather strict requirement from the PD side. Thus, changing Δx at fixed physical viscosity or equivalently Reynolds number, implies altering the numerical viscosity. If we utilize the method of matching advection, we require that the characteristic velocity remains unchanged, $u = (\Delta x/\Delta t)u_{\text{LB}}$, implying that numerical viscosity changes as

$$\nu_{\text{LB}} = \frac{\Delta t}{\Delta x^2}\nu = \frac{u_{\text{LB}}\nu}{u}\frac{1}{\Delta x}.$$

However, this simple scheme is inconvenient if the numerical mass diffusivity of solutes must be kept fixed at varying Δx. In fact, mass diffusivity is primarily a function of the interatomic distances, their mutual interactions, and the solvent viscosity, so that the numerical counterpart, expressed as $D_{\text{PD}} = D\Delta t/d^2$, where d is the interparticle distance, should be left unchanged as much as d. In summary, we are left with a varying Δx, with fixed Sc, Δt, and d.

To analyze the implications on the solvent viscosity, let us recast the Schmidt number as follows:

$$\text{Sc} = \frac{\nu}{D} = \frac{\nu_{\text{LB}}}{D_{\text{PD}}}\frac{\Delta x^2}{d^2}.$$

From this expression, we see that in order to keep Sc, Δt, and d unchanged at changing Δx, the LB viscosity must scale as $\nu_{\text{LB}} \propto 1/\Delta x^2$. This shows that Δx cannot grow too large, which can cause undermining numerical stability. Typically, ν_{LB} should stay above 0.005, although resorting to an entropic version of LB may loosen this constraint.

Overall, this strategy can achieve a significant speedup of the simulation, reaching a factor of 3 in solvent coarsening and a sizable speedup of 27 for the LB solver, without affecting protein diffusivity and related phenomena, such as the kinetics of peptidic aggregation.

1. The Boltzmann number

As discussed earlier in this review, the fluctuating LB (FLBE) is meant to account for the statistical fluctuations about the equilibrium state being dictated by the number of particles in the discrete states as is the case for actual molecules.

The relative intensity of the fluctuations, proportional to $1/\sqrt{\Delta N}$, is controlled by the so-called Boltzmann number, defined as (Dünweg and Ladd, 2009)

$$\text{Bo} = \left(\frac{\theta}{n\Delta x^3}\right)^{1/2}.$$

For most fluids $\theta \equiv v_T^2/c_s^2 \sim O(1)$, $\theta = 1$ corresponding to the ideal gas limit. The denominator is the number of molecules represented by a single LB particle, once we stipulate a number density $n_{\text{LB}} = 1$ in lattice units, which is always possible in an incompressible fluid.

The squared Boltzmann number can also be recast in the following alternative form:

$$\text{Bo}^2 = \theta \left(\frac{d}{r_0}\right)^3 \left(\frac{r_0}{\Delta x}\right)^3,$$

where d is the interparticle distance and r_0 is the range of the intermolecular potential. The second term on the right-hand side is the granularity parameter, controlling the degree of nondiluteness, while the third one is a direct measure of the spatial coarse graining.

By definition, the standard nonfluctuating LB corresponds to the macroscopic limit

$$\text{Bo} \to 0.$$

Note that in a liquid, where $d/r_0 \sim O(1)$, and even more so in a dilute gas, $d/r_0 \gg 1$, the smallness of the Boltzmann number is entirely in charge of spatial coarse graining, i.e., $r_0/\Delta x \ll 1$. In a nanofluid, however, the latter condition is no longer guaranteed.

For macroscopic flows, the Boltzmann number is pretty small indeed, typically of the order of the inverse square root of the Avogadro number.

For a millimeter cube of water, about 30 molecules per cubic nanometer: with $\Delta x = 10^{-3}$, $d/r_0 \sim 1$, and $\theta \sim 1$, we obtain $\text{Bo} \sim 10^{-10}$.

In microfluidics, the Boltzmann number gets larger: with $\Delta x = 10^{-6}$ m, we obtain $\text{Bo} \sim 10^{-5}$, still much smaller than 1, no point for the FLBE.

Nanofluidics, though, tells another story; with $\Delta x = 10^{-9}$ m, $\text{Bo} \sim 0.2$. By pushing the LB method even farther down, to atomistic scales, $\Delta x = 0.1$ nm (Horbach and Succi, 2006), the Boltzmann number may get even larger than 1: the LB particles become "quarky," i.e., they represent a fraction of a real molecule.

Clearly, this is a strongly fluctuating regime, for which the notion of the FLBE as a weak perturbation on top of the LBE comes under heavy question. In fact, at such subnanometric scales, the very notion of the distribution function becomes shaky because of many-body density correlations. The question is how can we possibly use the fluctuating LBE altogether at such small scales? Here mass and temperature come to some rescue as detailed later.

FLBE simulations work at $v_T^2 \sim 10^{-4}$–10^{-3}, much smaller than $c_s^2 \sim O(1)$, both in lattice units, for otherwise numerical instabilities arise due to the stochastic source being too strong.

This constraint is conveniently analyzed in terms of the "thermal mass" introduced by Dünweg and Ladd (2009), namely,

$$m_T \equiv \frac{k_B T}{c_s^2} = m\theta,$$

where m is the mass of the solvent molecules, i.e., $m/m_i \ll 1$.

Given that the inertial mass is typically set to $m = 1$ in LB units, if the thermal mass were to coincide with the inertial one, i.e., $m_T = m = 1$, one would indeed obtain $v_T^2/c_s^2 \sim O(1)$ in lattice units.

Since in actual practice the above ratio is of the order of 10^{-3} at most, the thermal mass is much smaller than 1, thus

ensuring the condition $\text{Bo} \ll 1$ even when fluctuations in particle number are of the order of $O(1)$.

In practical terms, it is like replacing a particle of mass m with $m/m_T \gg 1$ particles of mass m_T, a procedure with many similarities with variance reduction techniques popular in Monte Carlo simulations.

VI. SIMULATIONS AT THE PHYSICS-CHEMISTRY-BIOLOGY INTERFACE

In order to appreciate the specificity of the LBPD approach to tackle diverse problems at the PCB interface, we consider the computational complexity of the distinct LB and PD components of the methodology. The number of floating point operations needed to simulate a given problem over its characteristic dynamical evolution depends on the representation adopted for its constituents. In addition, $k = \#\text{Flops}/(\Delta x^3 \Delta t)$ is the computational density, being given by the ratio of the number of floating point operations needed to update the degrees of freedom contained in an elementary cell of volume $V = L^3$ and for a time T, both in lattice units. The complexity (Flops) is thus expressed as

$$\mathcal{C} = (k_{\text{LB}} + pk_{\text{PD}})L^3 T, \qquad (52)$$

where p is the fraction of particles contained in the elementary lattice volume Δx^3. Note that the cost of the LBPD cross coupling has been empirically absorbed by the prefactors k_{LB} and k_{PD}.

For diffusion-dominated applications, the typical case in biological processes $T \sim L^2$ thus $\mathcal{C} \sim L^5$, while for ballistic or convective dynamics, $T \sim L$ thus $\mathcal{C} \sim L^4$. Granted that for quantitative purposes, $\mathcal{C}$ needs to be evaluated for the specific application at hand, as a first estimate, we assume $k_{\text{LB}} \sim pk_{\text{PD}} \sim 10^4$.

A similar argument goes for the memory demand, which can be written as

$$\mathcal{M} = 8(n_{\text{LB}} + pn_{\text{PD}})L^3 \text{ bytes}, \qquad (53)$$

where n_{LB} is the number of discrete LB populations per lattice cell and n_{PD} is the number of degrees of freedom per discrete particle. For a standard single-species LB scheme $n_{\text{LB}} \simeq 20$, while for pointlike particles $n_{\text{PD}} = 6$ (position and momentum), for rigid bodies $n_{\text{PD}} = 12$ (position, momentum, angles, and torque) and for extended deformable bodies n_{PD} can reach up to several hundreds. For the applications to be discussed in the sequel, we simply take

$$\mathcal{M} \simeq 10^3 L^3 \text{ bytes}, \qquad (54)$$

which is an overestimate for dilute rigid particles and a likely underestimate for dense deformable ones.

A. Biopolymer translocation

The translocation of biopolymers, in particular, DNA or RNA strands, in nanometric pores provides a showcase of the synergistic hydrodynamic effects assisting or interfering with the translocation process. The translocation mimics a

genuinely biological one, whereby viral penetration takes place via the injection of viral genetic material into the host cell's cytoplasm. At a technological level, understanding how the physics of nanopores controls translocation inspires new paths to fast DNA sequencing (Fyta, Melchionna, and Succi, 2011). Nanopore-based technologies ultimately aim at translocating polynucleotidic chains through a nanoconfined environment, where the genetic information can be decoded by optical mapping, ionic, or electronic detection. The challenge is to control the process and the random, squiggly forms that the polynucleotide takes in solution, eventually designing nanofluidic devices according to stringent photolithographic requirements.

Computer simulations have the ability to access the fine details of the translocation process, both for technological innovations and for a better understanding of the biological processes involving the migration of small biopolymers. Consequently, various applications of LBPD have also appeared in recent years (Chen *et al.*, 2007; Hammack, Chen, and Pearce, 2011; Ledesma-Aguilar, Sakaue, and Yeomans, 2012; Hickey, Holm, and Smiatek, 2014; Alfahani, Antonelli, and Kreft Pearce, 2015).

Biopolymer translocation has been analyzed in different setups and modeling details for the translocating biopolymer, starting with a single necklace and neutral polymer threading between two chambers driven by a localized force acting only within the pore region (bead pulling) (Fyta *et al.*, 2006a; Fyta, Melchionna, Succi, and Kaxiras, 2008; Melchionna *et al.*, 2009). This setup mimicked real experiments where, given the presence of two electrodes at large distance from the pore, the electric field is overly intense where resistance is higher and mostly constant inside the pore region. In addition, electrostatic interactions stemming from the charged polymer are modeled in terms of effective beads of the chain. Simulations of translocation in small pores have thus focused on the dependence of the translocation time on the polymer length, thereby showing that the effect of hydrodynamic interactions is best seen on the translocation time versus chain length $T \propto N_b^{\alpha}$, with the characteristic exponent being 1.27 for short and 1.32 for long polymer chains, a result that is explained in terms of scaling analysis and energetic considerations that are peculiar to such a hydrodynamic-assisted process. Translocation in large pores showed an even richer phenomenology, with the appearance of several different configurations of the polymer folds, the consequence of fast translocation events that create discrete states that reflect on quantized current blockades on the measurable ionic currents (Bernaschi *et al.*, 2008; Melchionna *et al.*, 2009). Even more central is the role of electrokinetic forces on the process, the physical ingredient that can be included only by the full solution of the charged polymer whose translocation is driven by the self-consistent electric field. Even by including the double helix structure of polynucleotides, the complexity of the numerical apparatus can be optimally handled within the coherent LBPD framework, complemented by the solution of the Poisson equation for electrostatics. The result is a detailed description of translocation and the measure of the ionic currents, locally modulated by the threading polymer and being the result of the concurrent effects of excluded volume,

drag, and electrostatic forces (Datar *et al.*, 2017). Importantly, the development of new coarse-grained potentials for DNA (Hsu *et al.*, 2012) and RNA (Cruz-León *et al.*, 2018) paves the way to reveal the effect of the strong charging of the nucleic backbone that could not be elicited by using more aggressive coarse-grained models (Miocchi *et al.*, 2019).

For the translocation process, the associated threading time is proportional to the number of beads of the polymer N_b and in lattice units, and in lattice units it can be estimated as

$$T \approx 10^2 N_b^{1.27},$$

where the characteristic exponent is a direct signature of hydrodynamic interactions assisting translocation (Storm *et al.*, 2005; Fyta, Melchionna, Succi, and Kaxiras, 2008). Most of the computational time goes into solving hydrodynamics via the LB component, while the time to compute the mechanical forces and evolve the PD component is negligible. Consequently, for a typical size $L = 10^2$, the problem requires

$$\mathcal{C} \simeq 10^{13} \text{ Flops} = 10 \text{ TeraFlops}$$

to translocate a chain of about 100 beads. This is well within the capabilities of present-day computers. In fact, scaling up the size L by a factor of 10 would take to the order of Exaflops, still feasible on present-day leading-edge Petaflops/s computers.

As to memory requirements, based on Eq. (54), one estimates $\mathcal{M} \simeq 10^9$, a rather modest gigabyte. These figures reflect the fact of working with highly stylized polymers, without internal structure and chemical specificity (physics and biology instead of PCB).

The setup of translocation consists of two large chambers, a *cis* and a *trans* chamber, containing the pretranslocating and posttranslocated portions of the DNA strand. The chambers are typically much larger than the nanopore characteristic size. Translocating a long DNA or RNA chain into extremely narrow pores results in large entropy loss caused by the confinement and the need to stretch the macromolecule. The associated free-energy barrier reduces the biopolymer capture rates and causes clogging at the nanochannel or pore entrance. On the other hand, solvent-assisted interactions lubricate the process. It is key in understanding that the hydrodynamics of the translocating biopolymer in such fluidic device, being modulated by competing forces acting in the chambers and in the pore, give rise to a genuine multiscale scenario (Fyta *et al.*, 2006a, 2006b, 2006c; Melchionna *et al.*, 2007; Bernaschi *et al.*, 2008; Fyta, Melchionna, Kaxiras, and Succi, 2008; Fyta, Melchionna, Succi, and Kaxiras, 2008). When facing such a complex setup, all-atom molecular dynamics methods, or even coarse-grained representations of the translocating biopolymer, neglect the explicit representation of the solvent, thus imposing severe limitations to the overall accuracy. Resorting to strategies based on a direct solution of the NS equations, or using other mesoscopic numerical methods (Lagrangian or Eulerian based), is challenging in terms of generating consistent fluctuations under confinement and achieving a stable numerical method. In this respect, the

LBPD method is attractive because it allows generating the thermal fluctuations in a natural way and guarantees numerical stability over a wide range of translocation rates. In addition, one can analyze biomolecules of different size and initial configurations, in situations where the biomolecule is approaching the pore or is already in a docked configuration. LBPD has been utilized to analyze multiple scenarios (Fyta *et al.*, 2006b; Chen *et al.*, 2007; Fyta, Melchionna, and Succi, 2011) when the biopolymer has lateral size comparable or smaller than the pore diameter, conditions giving rise to single-file or multifile translocation configurations as illustrated in Fig. 2.

When accounting for the simultaneous presence of hydrodynamic and frictional forces, one can initially rely on the assumption of charge neutrality for the biopolymer and saline solution, a simplification justified by the need to reduce the computational effort. However, electrostatics is essential to guide the ionic currents and the current blockades caused by the impeding DNA molecule. A direct understanding of the ionic current blockades provides a stringent comparison with experimental measurements. The situation is even more complicated under flow conditions, whereby the interplay between electrostatics and flow does not allow one to utilize simplified solutions based on the assumption of global or local equilibrium. The inclusion of electrokinetics, that is, the representation of the multicomponent saline solution that flows together with DNA from chamber to chamber, provides direct access to the electrohydrodynamic process (Fyta, Melchionna, and Succi, 2011).

B. Ion channels

As anticipated, electrohydrodynamics is a fundamental aspect of the biological function, in particular, with regards to ion channels, the prototypical example of nanoscopic pores that subtend to the passage of ions in and out of the cell and regulate its volume. Ion channels are found within the membrane of most cells and are basically proteins that form the pore connecting the inner and outer parts of the cell. They look like narrow, water-filled pores that allow ions of certain types to pass through via selective permeability, privileging specific species, typically sodium or potassium. The transport of monovalent or divalent species depends crucially on the morphological properties of the confining elements that decorate the pore, notably charged peptidic groups that form the inner scaffold of the channel (Doyle *et al.*, 1998; Hille *et al.*, 2001).

Knowledge of the way that ionic transfer takes place unveils the biological functioning, but simulating a large biological aggregate composed of a membrane, ion channel and the inner and outer sides of the cell comprise a number of degrees of freedom, easily in excess of millions. Because of the large spread of relevant time scales, often inaccessible to today's computers. In principle, an alternative route is to leverage the statistical-mechanical approach such that the atomistic representation of the pore proteins is substituted by higher-level, coarse-grained descriptions. Another pillar of kinetic modeling, the Nernst-Planck equation, makes drastic simplifications by neglecting hydrodynamics altogether, but provides the

fluxes of ionic species as a function of the concentration and applied voltage. Such drastic simplification misses the fine details of ionic transport and the imperfect screening occurring inside the narrow cavities of the ion channel. From an operational standpoint, studying ionic transport requires matching the atom-based with the continuum-based description, a computationally unviable route due to the large space or time gap separating the two levels (Marconi, Melchionna, and Pagonabarraga, 2013).

As to translocating DNA, an optimal strategy is to proceed along the tandem LBPD path, whereby any feature that takes place at the fine atomic scale can surface up at the largest available scale, with its full content of long-range and unscreened electrokinetics. The numerical approach grants access to the characteristic ionic response by combining the fluid dynamics of multiple species in solution, the interplay of electrostatics and viscous forces, together with chemical specificity for the confining protein. The latter is particularly effective in determining the fine features within the pore lumen and vestibules that are responsible for ionic selectivity.

Another intriguing aspect of ion channels functioning is the fact that transport takes place under strictly microscopic confinement, whereby the competition among diffusive, stochastic, and migration forces together with the channel walls acts as an effective thermalizing bath for the moving ions.

Ion channels have provided a stringent benchmark to quantify the mechanisms by which local details arising from the channel geometry and the surface charge, the salinity of the electrolytic solution and the physical scale under study, affect ionic transport and the ensuing biological function. Electrokinetic forces have been shown to be highly modulated by geometrical details and by the channel surface charge (Melchionna and Marconi, 2011; Marconi and Melchionna, 2012). The presence of internal vestibules of the biological channel, for example, are easily modeled in the simulation and provide a direct inspection of the way that the electric field focuses along the channel axis, thereby modifying its activity. The role of axial asymmetries can be probed directly. Assimilating the channel shape to a conical one revealed the peculiar characteristic curves where currents are highly rectified by rather modest shape asymmetries. Similarly, the presence of boundary effects at the channel inlet are crucial to capture ions from the bulk and convey them under confinement by lowering *de facto* the involved energy barriers (Chinappi *et al.*, 2014). The effect of millimolar concentrations of electrolytes has been studied in terms of the double layer theory and revealed the role of screening on confined transport. Finally, and possibly most importantly, the role of nanoscale forces stemming from excluded volume interactions, acting among solvent molecules and ions, provides the critical ingredient to understand transport under strong confinement (Marconi, Melchionna, and Pagonabarraga, 2013).

Within the LBPD framework, let us estimate the computational effort by considering a typical current of 1 pA, traversing an ion channel corresponding to a flux of 10^7 ions per second. Consequently, to study the passage of a single ion in a simulation box of edge length $L = 10^2$ (accommodating a membrane of thickness 4 nm) with a time step of 10^{-12} s, required to cover 10^{-7} s, delivers a computational complexity:

$$\mathcal{C} \simeq 10^4 \times 10^6 \times 10^5 \text{ Flops} = 1 \text{ PetaFlops}.$$

This is well within reach of standard computational resources, a Teraflop computer would deliver in less than a hour. With $L = 10^2$, the memory requirements position around the tens of gigabytes.

However, such effort may not be needed to understand the ionic currents semiquantitatively. In fact, under confinement, hydrodynamics and long-range coherent motion of the aqueous solution dissipate away, due to the channel wall. Therefore, the continuum picture of fluid flow is not the most effective way to represent the ion channel or the entire embedding membrane in 3D. Alternatively, the Fokker-Planck equation describes well the action of the thermalizing channel wall. The mandate is then to cast the Fokker-Planck equation within the LB framework, a task that was successfully accomplished a decade ago (Melchionna, Succi, and Hansen, 2006; Moroni et al., 2006). Although less popular than its fluid dynamic counterpart, the lattice Fokker-Planck methodology shows the same levels of accuracy, robustness, and scalability.

The LB method can be comfortably extended to a broad variety of kinetic equations and one more proof comes from handling excluded volume interactions. The atomic correlations stemming from both electrostatics and excluded volume interactions are particularly intense under the channels operating conditions. Modeling correlations is a crucial element that other methodologies, such as the Nernst-Planck or dynamical density functional theory, cannot provide in conjunction with the solution of the NS dynamics.

The LB method applied to ion channels has found various applications, although most of them neglect the role of excluded volume and local specificity. Excluded volume forces acting between molecules can be determined starting from the Enskog collisional kernels, a revised version of kinetic theory of gases, by resolving the ballistics of hard-core collisions (Melchionna and Marconi, 2008; Marconi and Melchionna, 2009, 2011a, 2011b). The LB scheme accommodates this new collisional kernel in a natural way, another example of the versatility of the LB framework. An upsurge of more chemical-specific LB schemes is awaited for and expected in the forthcoming years.

C. Protein diffusion and amyloid aggregation

Mesoscopic simulations of macromolecules in aqueous solvent not only allow one to account for nanometric-scale hydrodynamics, but also for macromolecular interactions that are of paramount importance to avoid misfolding (Knowles, Vendruscolo, and Dobson, 2015) and molecular recognition. An important question is the extent to which molecular details are sufficient to reach the required level of biological realism. The answer is definitely problem specific: representing a protein, a DNA chain, or a lipidic chain may require different degrees of chemical specificity, depending on the research target in point.

It is also legitimate, however, to utilize coarse-grained force fields in a rather flexible way, as long as the mesoscopic properties, fixed at the nanometer or nanosecond scale, are reproduced.

Following these lines, simulations of 18 000 proteins have been performed to demonstrate the capabilities of the computational method, together with the parallel scalability on the Titan hardware platform composed of 18 000 GPUs (Bernaschi et al., 2013a), a multiprotein configuration being sketched in Fig. 3. Such a simulation allowed one for the first time to observe the diffusional properties of the solution under realistic crowding conditions. Another valuable application is provided by the joint usage of the particle-based approach with a multiphase LB scheme (Sega et al., 2013). A direct illustration of the nontrivial fluid-particle interplay in the formation and modulation of membranes driven by the action of drag and solvophilic forces is shown in Fig. 5.

To answer the question about the optimal scale to represent given biological solutions, this is where kinetic modeling, particularly for the liquid state solution, and the force fields match in accuracy.

At larger scales, micrometers and above, such level of detail may become irrelevant; therefore a fair representation for thermodynamics, possibly via an equation of state, and an accurate representation of fluid mechanics, may fulfill most practical needs. The scenario should also cope with the possible action of long-range forces, especially of electrostatic origin. Fortunately, cellular conditions are such that in bulk conditions and away from the compartment boundaries, screening acts as a powerful localizer of interactions that die off at distances above a few nanometers.

In order to integrate the protein force fields with the physicochemical features of solvation, the LB framework should also be enriched with waterlike features, inclusive of directional interactions, and hydrogen-bond features, having deep implications on the macromolecular structures (Papoian et al., 2004). Preliminary efforts in this direction have been made in the past, but their thorough validation remains entirely open (Succi et al., 2014).

Many applications of a more water-specific methodology naturally suggest themselves: the thermal stability of proteins, the onset of neurodegenerative diseases due to peptidic aggregation, the diffusion of proteins, and trafficking in cellular crowding, being just some examples in point.

Besides initial foot-in-the-door applications, this plan requires a massive amount of implementation and validation work, one that possibly suggests the need for coordinated community efforts.

The aggregation of misfolded soluble proteins into fibrils is the precursor of several neurodegenerative diseases, such as the Alzheimer, Parkinson, and Huntington ones. In particular, Alzheimer's disease is marked by atrophy of the cerebral cortex showing accumulation of amyloid plaques and numerous neurofibrillary tangles made of filaments of the phosphorylated tau proteins. The major constituents of plaques are made of the amyloid β peptides made of 40 and 42 amino acids (Buckner et al., 2005).

The fibrillogenesis of amyloid β peptides is a complex process whereby fibrils extend up to hundreds of nanometers, and the time scale of full growth exceeds hours in vitro. The details of the emergence of amyloid protofilaments are still debated but it has been observed that the formation of ordered arrays of hydrogen bonds drives the formation of β sheets within aggregates that form early under the effect

of hydrophobic forces (Auer *et al.*, 2008). Understanding the mechanisms of amyloid aggregation is key to the design of drugs able to prevent fibril formation and toxicity in the brain and computer simulation is an essential tool to explore the aggregation process. First and foremost, describing the kinetics of amyloid formation via conventional nucleation theory lacks information on the structure and size of the primary nucleus.

Mimicking amyloid aggregation cannot be done by implicit solvent models, since the lack of solvent interactions does not include the treatment of solvation thermodynamics and altogether neglects the action of solvent-mediated correlations. Self-assembly initiates via a hydrophobic collapse and the formation of molten oligomers, with the common feature of fibrils being the interdigitation of the side chains, the so-called steric zipper. Since fibril formation is under kinetic and not thermodynamic control, this is a great showcase for the LBPD strategy (Nasica-Labouze *et al.*, 2015).

Large-scale aggregates form only with the correct kinetics and show up the correct intermediate and metastable states by using the highest level of physical fidelity. If a simplifying assumption on the dynamics, such as those provided by the Langevin level, is used, spurious intermediate states eventually kick in, as long as the final aggregate is kinetically driven.

By including hydrodynamic interactions and by employing the OPEP force field, the LBPD methodology has shown that the solvent-mediated interactions have a key role in regulating amyloid aggregation; see Fig. 4 (Chiricotto, Melchionna *et al.*, 2016; Chiricotto *et al.*, 2017). As a matter of fact, hydrodynamics enhances peptidic mobility, thus facilitating mutual encounters and collapse of the aggregated structure. This should be appreciated in view of the nontrivial computational effort required to simulate the aggregation process, which is not only driven by diffusion but also features a slow down due to the energetic barriers involved in the process. For a dense number of peptides, $p_{PD} \simeq 1$ and a typical box edge of length $L = 10^2$, the complexity is about 2 orders of magnitude larger than a purely diffusive process, resulting in a total of

$$\mathcal{C} \simeq 10^2 \text{ TeraFlops.}$$

With current hardware facilities, the aggregation process can be observed during its full course. This is an important, possibly first real-life application of the methodology, in a situation where macromolecular realism and solvent-mediated interactions change drastically the conventional picture of amyloid aggregation as compared to assuming negligible hydrodynamic forces (Sterpone *et al.*, 2014).

This becomes even more interesting when looking at the effect of shear flow onto the kinetics of aggregation. In a Couette flow, fibril formation can accelerate from one month down to a few hours (Dunstan *et al.*, 2009; Bekard *et al.*, 2011).

A possible mechanism for the effect of shear is the alignment of aggregates, which in turn facilitates their assembly. Even changing the specific nature of the shear flow can enhance the formation of protofibrils and the growth of fibrils. Clearly, there is still a long way to go toward a full

characterization of the aggregation process, but at this point, it is clear that the LBPD strategy is highly apt at coping with this complex scenario.

A side observation regards the computational efficiency of LBPD to simulate pepditic solutions. Clearly, interparticle interactions are a major bottleneck, due to the burden of computing a large number of nonbonding and bonding forces, with a significant share of computing a large number of nonbonding and bonding forces, with a significant portion of computing time spent in searching interacting pairs and bookkeeping them. In addition, as seen earlier, chemical realism requires one to account for rather stiff forces and therefore a consequent small time step imposed on the particle solver. If the system is dense in particles, handling interparticle forces is going to be the slowest segment of the simulation, while the optimal setting is when particles are in diluted or semidiluted conditions. This is precisely the operating conditions pertaining to the aggregation of amyloid β peptides.

The effect of hydrodynamics on protein diffusion has been studied for a solution made of 18 000 Rat1 proteins in a bulk simulation at 40% volume concentration (Bernaschi *et al.*, 2013a; Sterpone *et al.*, 2014). The study showed that in such crowding conditions, protein diffusion proceeds according to the experimental values measured by quasielastic neutron scattering and pertaining to the 3.5–5 ns temporal range, exceeding the hydrodynamic time scale arising from the propagation of vorticity over the protein linear size, and ultimately slowing down the protein self-diffusion. A drop of the diffusion coefficient at volume fraction between 10% and 30% marks the onset of caging mechanisms, whereas at larger volume fractions the diffusivity dangerously approaches a jamming transition, while protein motion is still in action. Amyoid aggregation represents an examplar instance of hydrodynamic forces impacting the formation of molecular aggregates. By studying a system of unprecedented size, LBPD simulations were able to explore a branched disordered fibril-like structure that had never been described by computer simulations before (Chiricotto, Melchionna *et al.*, 2016). The results show that hydrodynamics forces also steer the growth of the leading largest cluster and impact the aggregation kinetics and the fluctuations of the oligomer sizes, by favoring the fusion and exchange dynamics of oligomers between aggregates.

D. Toward computational physiology and medicine

Physiological flows offer one of the most attractive applications of the LB framework to real-life situations, with high potential social impact in utilizing computer simulations to diagnose pathologies, prognose a medical condition, or even guiding clinical intervention (Fenner *et al.*, 2008; Noble, 2008, 2016; Patronis *et al.*, 2018).

In the age of evidence-based medicine, the decision-making process needs to be optimized by using evidence from well-designed and well-conducted research. Although all medicine has some degree of empirical support, the evidence-based approach requires that only the strongest data coming from meta-analyses, systematic reviews, and randomized controlled trials can be used to inform clinical recommendations.

Incidentally, this spawns major opportunities for the synergistic operation with machine-learning techniques (Goldberg and Holland, 1988), a delicate subject we will return to at the end of this review.

Physiological flows, for instance, are conditions where a biofluid circulates in complex anatomical conduits and networks, examples ranging from blood flow to lymphatic circulation, to airways, the urinary system, etc.

Blood flow has made the subject of intense research over the last decades, the application of LB to blood flows experienced a major burst of activity (Buick *et al.*, 2003; Dupin, Halliday, and Care, 2003, 2006; Sun, Migliorini, and Munn, 2003; Sun and Munn, 2006; Boyd, Buick, and Green, 2007; Sui, Chew, and Low, 2007; Dupin *et al.*, 2008; Hua-Bing, Li, and Bing, 2008; Sui *et al.*, 2008; MacMeccan *et al.*, 2009; Wu and Aidun, 2010; Reasor, Clausen, and Aidun, 2012; Xiong and Zhang, 2012; Krüger *et al.*, 2013; Shi *et al.*, 2013; Xu, Tian, and Deng, 2013; Krüger, Holmes, and Coveney, 2014; Pontrelli *et al.*, 2014), with several applications to coronary, carotid, and cerebral blood flow. Of course, this is no surprise, since these are macroscale applications for which the most conventional NS hydrodynamics appears perfectly adequate.

It is to be stressed that statistical averaging is of little meaning in a physiological context, since each individual is a story of her or his own. Rather, the detailed access to the patient-specific 4D hemodynamic data across scales of motion may offer a quantum leap in the quality and accuracy of preemptive medicine (Artoli, Hoekstra, and Sloot, 2003; Hirabayashi *et al.*, 2004; Pelliccioni, Cerrolaza, and Herrera, 2007; Caiazzo *et al.*, 2009; Rybicki *et al.*, 2009; Melchionna *et al.*, 2010; Groen *et al.*, 2013; Yun *et al.*, 2014; Omori *et al.*, 2015).

This is why a fully 4D (three-space dimensions and time) real-time numerical and visual access at the blood dynamic flow patterns from microns all the way up to the full-scale geometry can disclose unprecedented opportunities for personalized and precision medicine; see Fig. 15.

Again, a few numbers may help to convey a concrete sense of what is meant here. Based on Eq. (52), the computational complexity of a 4D real-time LBPD simulation covering four spatial decades ($L = 10^4$) and just a single circulation time ($T \propto L$) is of the order of

$$\mathcal{C} \sim 10^4 \times 10^{16} \text{ Flops} = 10^2 \text{ ExaFlops.}$$

This is at least a factor of 10 short of the real time span needed to collect significant diagnostics, so let us take 1000 Exaflops (1 Zettaflop) instead. On a current-time Petaflop computer, this makes 10^6 seconds wall-clock time, about two weeks, which is not exactly what one would label as real time. A prospective Exaflop computer, though, would complete the job in some 20 min, thus bringing the real-time task within direct clinical fruition.

LB simulations of physiological conduits offer an exciting opportunity due to its most practical asset: simplicity in handling complex geometries and in automated mesh generation. To the best of our knowledge, such simplicity remains unparalleled as compared to grid methods for the numerical
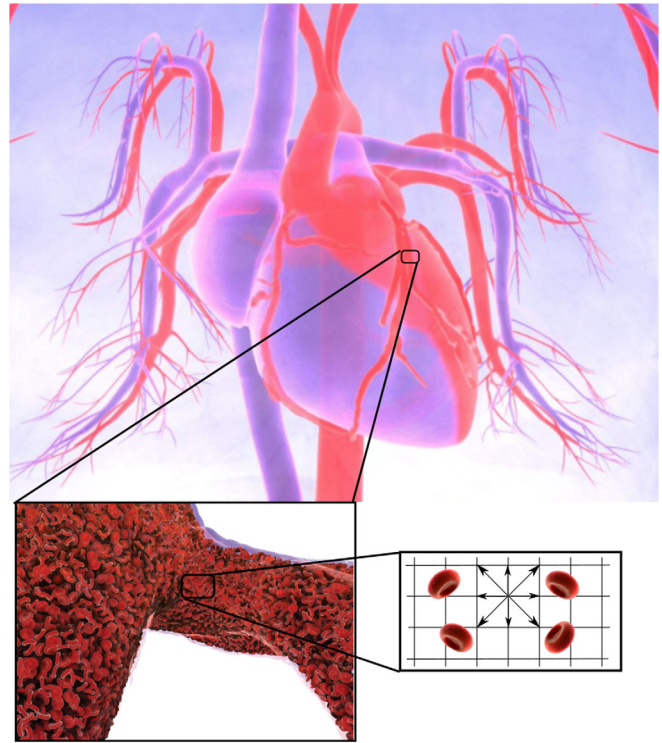

FIG. 15. Three-level representation of blood flow. Upper panel: continuum macroscale (cm). Bottom left panel: the granular nature of red blood cells starts to be apparent (100 μm) and is fully revealed at the level of the cell spacing (10 μm, bottom right panel). From Bernaschi *et al.*, 2013b.

solution of NS equations. Another strength of the LB simulation comes from its local structure in space and time. As typical in the study of unsteady flows, the flow patterns are particularly rich and accurate once unsteadiness is taken into account as for the study of gas flows. Hemodynamics also sets a case, due to the large excursions of local Reynolds number (going from virtually zero in microcapillaries to nearly 10 000 in the aorta), whereby unsteadiness promotes both local and global patterns. Most importantly, blood is pumped into the vessels by a pulsatile injection rate, a situation that requires time-explicit boundary conditions. In biomechanics, the ratio of transient inertial to viscous forces, the Womersely number, can range from 10^{-3} in capillaries to 10 in the aorta, calling for the direct time-explicit solution in the most general case. The applications are widespread, but it is worth recalling the study of coronary and carotid arteries, as critical vessels that subtend to the oxygenation of the heart muscle or the brain. Any anomaly in the blood flow would cause major risks of heart attack or stroke to the patient. In such networks of arteries, geometric complexity is highly nontrivial; particularly challenging is the handling of conditions where narrowings and plaques give rise to eccentric passages and tiny spaces, at times as large as a handful of red blood cells.

The way pressure is distributed in arteries has thus great physiological relevance, since the supply and demand of oxygen in organs, primarily the heart muscle, is regulated by the distribution of pressure in vessels. It is well known that narrowings and blockages lead to strong pressure losses, with

consequent starvation of the tissues depending on blood circulation.

Here the LB method has a technical, yet important point. When looking at the possible buildup of atherosclerotic plaques, the LB framework offers facilitated access to the wall shear stress tensor. This can be computed *locally* based on the nonequilibrium populations, via

$$\overset{\leftrightarrow}{S}(\vec{r}, t) = \frac{\nu\omega}{c^2}\sum_i \vec{c}_i\vec{c}_i(f_i - f_i^{\text{eq}})$$

thereby dispensing with expensive and inaccurate finite-difference operations.

As mentioned several times in this review, a truly major asset of the LB method is parallel performance, with a large potential for the biofluidic context. As in other applications, the LB mesh is plain Cartesian and, for most biofluidic applications, a single resolution is sufficient. Because of the excellent scalability of the basic method, one may suppose that parallel performance would persist also for intricate vessel networks. Unfortunately, in this case the computational domain covers a sparse region of space, typically filling space by less than 10%. This is definitely a challenge, even for a highly scalable LB solver and countermeasures have to be taken. Possible solutions to this issue are discussed in the Appendix.

The LB method is today widely used to study biofluidics and blood flow. The fast turnaround makes it an excellent candidate for rapid screening in clinical practice, with the prospective possibility to even predict the outcome of clinical intervention. Stenting, angioplasty, flow diverting, and aneurism wiring, to mention but a few, are all possible applications of the LBPD methodology. Clearly, this is not the end of the story, as moving parts, as concerning valve placement or compliant vessel deformations can be taken into account by using the various schemes available today, starting from the IBM to handle the moving walls (Fang *et al.*, 2002; Hoekstra *et al.*, 2003; Descovich *et al.*, 2013; De Rosis, 2014).

Microcirculation is interesting in itself. Because of the red blood cells constituents of blood that reach up to 45% in volume in humans, two main concurrent effects take place: the Farhaeus-Lindqvist effect, whereby the average concentration of red blood cells decreases as the diameter of the containing vessel is decreased; the second effect is the viscosity change with the diameter of the vessel it travels through. The two effects arise because red blood cells move preferentially over the center of the vessel, leaving the plasma at the wall, thus lowering the near-wall dissipation effects. The Farhaeus-Lindqvist effect becomes visible in the range between 10 and 300 μm.

In recent years, the study of microcirculation witnessed an upsurge of interest by using mesoscale particle methods (Noguchi and Gompper, 2005; McWhirter, Noguchi, and Gompper, 2009; Clausen, Reasor, and Aidun, 2010; Fedosov, Caswell, and Karniadakis, 2010; Janoschek, Toschi, and Harting, 2010; Melchionna, 2011; Pontrelli *et al.*, 2012, 2014, 2015Matyka, Koza, and Mirosław, 2013). In capillaries of lateral size of 100 μm and below, the motion of red blood cells reveals highly nontrivial signatures of

granularity and deformability. For capillaries with a diameter of a few μm, erythrocytes undergo large deformations in order to squeeze into the vessel and the globules are able to crawl into the micrometer-sized space. On the other hand, when looking at larger-scale circulation, in the 100–500 μm range, it is generally sufficient to consider blood cells as rigid bodies. The grand challenge here is to reach up to physiological scales (1–10 cm) while retaining essential microfeatures, the finite size of red blood cells (8 μm) in the first place. This is of major interest for many reasons; the granular nature of blood may have a significant impact on the recirculation patterns in the proximity of natural geometrical irregularities, such as bifurcations, stenoses, aneurysms, or man-made ones, like stents and other medical devices.

Micro-to-macrohemodynamics is particularly rich, showing a peculiar distribution of oxygen-carrying cells at every bifurcation depending on the local Reynolds number, and with far reaching consequences on physiology. Erythrocytes exhibit both a tumbling motion and the tank-threading effect, whereby the cell membrane can slide under a shear force (Keller and Skalak, 1982), two conditions that have deep impact on blood rheology. Plasma skimming near the arterial walls has important consequences on the local and global circulation in order to optimize the oxygen supply chain, keeping, at the same time, the flow speed high in the capillaries. Further consequences relate to the most common of cardiovascular diseases since atherosclerosis depends on the uptake of lipidic material by the arterial wall and, ultimately on the near-wall shear stress. The discussion is still open and its outcome is extremely important to understand the causes of myocardial infarction for diagnostic or preemptive medicine.

Cellular hemodynamics is an open branch of research. A direct extension of the LBPD method can account for suspended bodies for the explicit presence of cells suspended in plasma. This is a typical case where the hydrodynamic medium hosts particles with finite size, anisotropic shape, in fact oblate ellipsoids that represent red blood cells to first approximation. Diverse community software packages (Heuveline and Latt, 2007; Mazzeo and Coveney, 2008; Bernaschi *et al.*, 2009; Clausen, Reasor, and Aidun, 2010), such as OPENLB, MUPHY, or HEMELB, are making their way to offer several cell-type capabilities and performance so much so that studying flows composed by red blood cells and leukocytes becomes extremely attractive. Previous work showed the complex hydrodynamic interplay between cells of different shapes, with the margination of leukocytes and their rolling along the vessel wall (Munn and Dupin, 2008).

All LB assets show great value for deployment in hemodynamics.

To appreciate the computational complexity of a typical blood flow system, let us consider a coronary arterial tree and estimate the number of mesh voxel needed to fill the sparse volume occupied by the vessels to 10^7. The problem is typically advective and the number of time steps to cover a pulsatile cycle is $\sim 10^6$. When red blood cells are also simulated, the computational effort on the PD side can easily exceed the LB component by 1 or 2 orders of magnitude. Consequently, ranging from LB to LBPD simulations of a coronary tree requires

$$\mathcal{C} \sim 10^2\text{--}10^4 \text{ PetaFlops.}$$

As discussed previously, with Exaflops computers at hand, this may be brought close to the realm of real-time simulation.

Besides the most appealing features, the downsides of the LB method should be mentioned also. An important point concerns low-Mach flow circulation. The virtually incompressible nature of blood flow (with $\text{Ma} < 10^{-3}$) can negatively impact the accuracy of simulations, in particular, when the pressure distribution is the goal of the investigation. The compressibility error of the LB method scales as Ma^2 and making it negligible typically requires lowering the simulated Mach number Ma^*, or equivalently, the numerical typical velocity, since $u/c = \text{Ma}^*$. However, the price to pay is a corresponding reduction of the LB time step, down to values that can easily be as small as microseconds, potentially undermining its practical efficiency for simulating biofluidics. The answer to the conundrum comes from practice and from the observation that, even at such low time step, computing efficiency is sufficient to resolve highly complex fluid flows. It is worth noting that the optimal working conditions can be problem specific, and a study versus resolution and simulated Mach is recommended.

Besides blood flow, the LB method applied to biofluidics is also witnessing medical applications to airways, ranging from nasal to pulmonary flows (Freitas and Schröder, 2008; Eitel *et al.*, 2010; Krause, 2010; Li and Kleinstreuer, 2011; Lintermann, Meinke, and Schröder, 2011; Miki *et al.*, 2012). This is yet another story, where compressibility effects become much less important and the focus shifts toward very intricate geometries in the presence of collapsible walls. Understanding how air flows in these regions, the consequence of rather small but crucial imperfections, or the transport of small molecules or odorants are all applications that draw great profits from the possibility of solving fluid mechanics in a multiphysics scenario in order to study peculiar conditions. Again, the LB and LBPD methods are particularly well suited to computationally embrace the presence of multiple agents in solution.

VII. SIMULATIONS AT THE PHYSICS-CHEMISTRY-BIOLOGY INTERFACE: LOOKING AHEAD

The growth in computing power that is expected to be sustained for the next decades will spawn tremendous opportunities to gain new insight into a series of fundamental problems dealing with complex states of flowing matter in general, and, in particular, those relevant to biology and medicine.

For reference, let us speculate what can and hopefully will be possible once LBPD operates at Exascale performance. Since this review is mostly intended to discuss applications at the physics-chemistry-biology interface, hereafter we will focus our attention mostly on that paramount scenario.

To that purpose, we reiterate that the LBPD description of biological systems is based on a mesoscopic picture, whereby molecular details are incorporated within suitable coarse-grained terms in the effective kinetic equation for the solvent and coupled to stochastic particle models of the biological molecules. The art, as usual, is to incorporate the least amount of molecular details required to describe the essential physical phenomena under scrutiny.

A. Future challenges: Toward extreme LBPD computing

The inclination of the LB method for parallel processing comes from the fact that within the LB formalism information always travels along straight streamlines, regardless of the physical complexity of the emergent flow structure. This marks a major divide versus macroscopic formulations, whereby information moves along material lines defined by the space-time dependent flow field $\vec{u}(\vec{r}; t)$. The point is key to achieve outstanding parallel efficiency also in the case of geometries having real-life complexity, such as those that often occur in biological problems where shape and function are tightly correlated, as described in previous sections. Albeit highly technical, this point is absolutely crucial to achieve the levels of extreme scalability [extreme LB (XLB)], which are mandatory to access the "disruptive" applications described in the previous sections. In a way, we may compare these technological advances to the development of a new experimental technique or device aimed at exploring new states of matter: tera-electron-volts in high-energy physics or teramolecules in computer explorations at the physics-chemistry-biology interface. Needless to say, scaling up to millions and soon billions of computing cores surely does not come for free, especially when the geometry is not regular; it must be won via very advanced and dedicated programming strategies [see the Appendix and the recent prospective paper by Succi *et al.* (2019).

However, once such efforts are put into place, the results are extremely rewarding on virtually any parallel platform. In the last decade a few multiscale codes coupling LB for the fluid motion with various forms of stochastic particle methods for the dynamics of floating bodies within the flow have been developed (Bernaschi *et al.*, 2009; Feichtinger *et al.*, 2011; Rossinelli *et al.*, 2015). Among these, MUPHY is a fully scalable LBPD code which has been successfully used for the simulation of a variety of biofluidic applications (Bernaschi *et al.*, 2009, 2011, 2013a). These include biopolymer translocation, multiscale hemodynamics and, lately, proteins. MUPHY has attained fairly impressive parallel performance, with an escalating progression from 11 TeraFlops/s for biopolymer translocation on the IBM Jugene (2011), to 0.7 PetaFlops/s for multiscale hemodynamics on Tsubame (2012), up to a world record (to the best of our knowledge) of 20 PetaFlops/s (sustained performance) for protein crowding on Titan (2013). Although such figures refer to leading-edge supercomputing experiments rather than fully fledged biofluidic applications, they point to a tremendous potential for prospective applications at the physics-chemistry-biology interface, suggesting to proceed along the road map illustrated in Fig. 16. However, extracting such potential on upcoming Exascale architectures faces a number of challenging issues. Since a successful handling of these issues is key to open up new and otherwise inconceivable LBPD applications at the PCB interface. In the Appendix we will provide a relatively detailed coverage of the main technical topics, including (i) how to best exploit modern CPU and accelerators architectures for LBPD simulations, (ii) how to exploit at their best
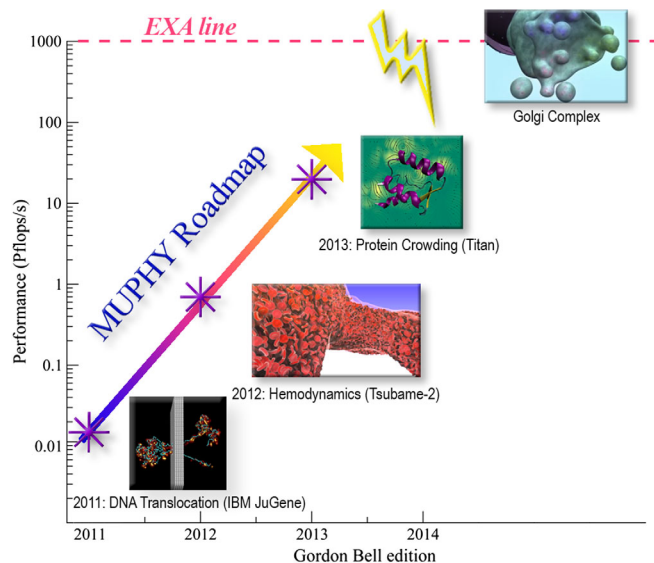
FIG. 16. The road map of the MUPHY code, based on the Gordon Bell performance obtained over the years. From Bernaschi *et al.*, 2013b.

modern CPU and accelerators architectures for the simulation of LB methods, (iii) how to overlap computation and communication, so as to hide the overheads of the latter, and (iv) how to secure a balanced load among millions of computing cores in realistic geometries such as the cell or blood vessels.

Having laid down the two-legged engine for fluids and molecules, a stringent requirement is to design accurate kernels to exchange forces among fluids and particles of different kinds: hydrodynamic, solvation, dispersion, electrostatic, etc. This program requires accessing algorithmic methods from a vast array of options and organizing various types of biological agents, representing proteins, nucleic acids, lipids, sugars, etc. At the core, one must devise new algorithms that allow reproducing environments of increasing complexity, either from the bottom-up molecular route or from the top-down macroscopic or mesoscopic perspective. Most importantly, the plan requires not only the representation of multiple agents (fluids, macromolecules or small molecules, continuum-based solutes) but also the architected orchestration of their evolution, as they move across multiple scales. This presents an outstanding challenge in both conceptual and technical respects.

At the time of this writing, a fully fledged multiscale numerical strategy is still lacking, thus making the subject of intense research. Since multiple particle-based and field-based agents need to coexist within a unified computational framework, the search for a robust and universal strategy to achieve their seamless coexistence is still pretty open.

The first step forward is to devise numerical methods whereby only the fluid component is allowed to cross scales: generally speaking, these fall in the class of multigrid fluid dynamics, where adaptive, multiresolution or body-fitted meshes are used to resolve fluid dynamic patterns via static or automatic mesh generation. Although pretty laborious, these methods have been developed for several decades and

are presently in the position of providing dramatic savings in both memory and number-crunching requirements.

A second, much more challenging aspect regards the evolution of multiple Eulerian or Lagrangian *transmuting* agents, meaning by this that such agents are capable of changing their identity and representation on the fly, depending on the local physics in point.

By its very mesoscopic nature, the LB method is conceptually at a vantage point for multiscale or level coupling, both upward, toward continuum representations and downward, toward atomistic models. For all its conceptual appeal, such a high-level program requires a concrete computational substantiation in order to turn theoretical ideas into actual computational tools.

A most useful LB asset in this respect is the Cartesian mesh. Owing to its simplicity, mesh construction, management, and extensions offer a particularly flexible approach to handle complex biological settings. As two or more neighboring scales are juxtaposed in space and time, mesh refinement corresponds to increasing the number of mesh points in proximity of geometrical variations, where mass or momentum gradients are most likely to attain peak values.

To date, a number of multigrid LB schemes have been proposed, some of which are customarily used in large-scale academic codes (Bernaschi *et al.*, 2009; Lagrava *et al.*, 2012) or in industrial applications (Chen *et al.*, 2003). The common approach is to consider two juxtaposed meshes, a factor of 2 ratio in spacing. A possible strategy is to exchange information between two neighboring meshes that overlap in some finite region of space and to exchange populations by spatial and temporal interpolation schemes. This hand-shaking procedure must make sure that not only the flow fields, mass, and momentum density are continuous across the surface, but also their fluxes. This leads to a specific map between the discrete populations in the coarse and fine grids. Another approach is to consider the fluxes across neighboring meshes by using a finite-volume description of the LB method and to exchange these components (Chen *et al.*, 2006). Yet another, recent but promising approach is to merge the standard LB formulation with unstructured finite-volume LB formulations in correspondence with sharp features of the flow (Di Ilio *et al.*, 2017).

Given the underlying mesh connectivity (e.g., by taking the D3Q19 mesh as a reference), the mesh nodes that are connected by a complete (18) set of mesh neighbors of the same mesh spacing are called "saturated" nodes as opposed to the "unsaturated" ones, which have only an incomplete (< 18) set of mesh neighbors connecting nodes from different meshes.

For clarity, let us define $\mathcal{S}^1, \mathcal{S}^2, \mathcal{S}^3, \ldots$, as a sequence of *scopes* that describe the ownership of fluids and particles to a single scale and help to coherently organize the computation across scales.

By construction, each scope accommodates a single Cartesian mesh $\mathcal{M}^1, \mathcal{M}^2, \mathcal{M}^3, \ldots$,

$$\mathcal{M}^k \in \mathcal{S}^k.$$

A given scope $\mathcal{S}^k$ contains a set of fluids $\mathcal{F}^{k,\alpha}$ and a set of Lagrangian particles $\mathcal{P}^{k,\alpha}$:

$$\sum_{\alpha}\{\mathcal{F}^{k,\alpha}\} \cup \{\mathcal{P}^{k,\alpha}\} \in \mathcal{S}^k.$$

By associating a fluid to a scope, by design, we posit that LB fluids are forced to belong to a given Cartesian mesh and cannot cross the interface between different meshes. On the other hand, since particles are, by definition, grid free they are allowed to move anywhere in space. As a result, their ownership to a given scope is purely a matter of organization.

However, particles can change their physical identity or even disappear (change to class "nihil") as they cross scales. In fact, a particle belonging to a physical domain can swap to another particle type (possibly with a different representation) as it crosses scale boundaries, without necessarily leading to a discontinuous trajectory.

As a further development, one can transform Lagrangian agents into Eulerian ones and vice versa, a procedure which involves not only interpolation but also projection from particles to LB distributions via hydrodynamic fields and direct sampling of particle coordinates directly from the LB distribution.

These "transmutational" functionalities raise genuinely new issues, both conceptual (compliance with the basic principles of statistical physics) and computational, i.e., the design and efficient manipulation of the corresponding data structures. It is conceivable that they might spawn the needed for dedicated transmutational multiscale-multilevel software as well.

Coming back to our conceptual scheme, meshes are best organized in a hierarchical order, such that the kth level mesh $\mathcal{M}^k$ has spacing Δx^k ordered in an ascending sequence $\Delta x^k = 2\Delta x^{k-1} = \cdots = 2^{k-1}\Delta x^1$, thus the mesh spacing doubles at each level increase.

Juxtaposed meshes feature a finite overlapping or compenetration between neighboring meshes (Lagrava *et al.*, 2012), and the extent of the compenetration region is taken as $L(\mathcal{M}^k \cap \mathcal{M}^{k-1}) = \Delta^k$.

In its simplest implementation, all boundary conditions operate on the finest mesh, covering regions where the strongest gradients need to be resolved, typically in the proximity of solid walls. In multigrid LB, let us consider the propagation in time of the LB fluid across a set of scales labeled $0, \ldots, K$ and illustrate the scheme based on interpolation or extrapolation of fluid populations rather than on the fluxes.

To that purpose, spatial averaging of a generic field A living on mesh $\mathcal{M}$ is given by

$$\bar{A}(\vec{r}^k, t) = \frac{1}{q}\sum_{p\in\mathcal{N}} A(\vec{r}^k + c_p^k \Delta^{\min}, t), \qquad (55)$$

where $\mathcal{N}$ is the set of all the nodes of M that are neighbors of the nodes of $\mathcal{M}'$ at $\vec{r}_k$, $q = \text{Card}(\mathcal{N})$, and $\Delta^{\min}$ is the smallest spacing between $\mathcal{M}$ and $\mathcal{M}'$. When $\mathcal{M}$ is finer than $\mathcal{M}'$, averaging or interpolation corresponds to a low-pass filter.

Another useful tool is time averaging between the latest updated time of $\mathcal{M}$ ($t_i^{\mathcal{M}}$) and the previous one ($t_{i-1}^{\mathcal{M}}$), which is needed to estimate the field A at time $t = t_i^{\mathcal{M}} = (t_{i-1}^{\mathcal{M}} + t_i^{\mathcal{M}})/2$:

$$\bar{A}(\vec{r}^k, t) = \tfrac{1}{2}[\bar{A}(\vec{r}^k, t_i^{\mathcal{M}}) + \bar{A}(\vec{r}^k, t_{i-1}^{\mathcal{M}})].$$

In this equation, the quantities $\bar{A}$ within the square brackets can be evaluated from Eq. (55).

The multigrid LB method proceeds by a local time-stepping frequency and by rescaling the relaxation time according to the local mesh in action. Each scope agent is evolved according to a specific propagator. For either LB fluids or particles, the evolution time step reflects the ownership to a given scale, with its own specificity. For the LB method, mesh spacing and time step are strictly related so that in a single streaming operation, information hops between neighboring nodes of the same mesh. For particles, the time step is an independent quantity and, as long as the energies involved are well sampled and the simulation is stable, one can tweak the time step to achieve optimal performance.

When information travels across neighboring meshes, the lattice speed should not change between meshes and fluid velocity and pressure should be continuous across the interface. Given a reference length scale ℓ and time scale τ, all other scales are such that $\ell^k = \Delta x^k \ell$ and $\tau^k = \Delta t^k \tau$. For the time marching, it is possible to adopt the convective scaling, which leaves the flow speed invariant across scales, i.e., $u^k = u^{k-1} = \cdots = u^1$.

Then the time step Δt^k is related to Δx^k via

$$\frac{\Delta x^k}{\Delta t^k} = \frac{\Delta x^{k-1}}{\Delta t^{k-1}} = \cdots = \frac{\Delta x^1}{\Delta t^1}.$$

By matching the Reynolds number across all scales to a single reference value $\text{Re} = u\ell/\nu$, where u is a typical flow speed,

$$\frac{u^k \ell^k}{\nu^k} = \cdots = \frac{u^1 \ell^1}{\nu^1} \equiv \text{Re},$$

we obtain the scaling relation for the fluid viscosity, namely,

$$\nu^k = 2\nu^{k-1} = \cdots = 2^{k-1}\nu.$$

Given the reference viscosity ν, a scale-specific frequency ω^k is derived. In fact, in order to have a single global kinematic viscosity, each species has the same viscosity ν, related to the relaxation frequency via Eq. (24).

It is important to appreciate the fact that the local nature of the LB methodology makes the construction of hydrodynamic moments and gradients, such as the deviatoric stress, for multiresolution as simple as in the single-resolution case. This marks a plus as compared to NS schemes, where gradients need to be computed through stencils or other numerical templates.

However, when considering embedded particles, a number of adaptations have to be catered to. The single mesh described in Sec. IV requires interpolation or extrapolation schemes that are well posed in terms of smoothness and robustness. The scheme then adapts to the situation where particles cross mesh (or scale) boundaries.

A particularly desirable extension to simulate biosystems concerns the case when the mesh locally readapts to accommodate the presence of moving macromolecules.
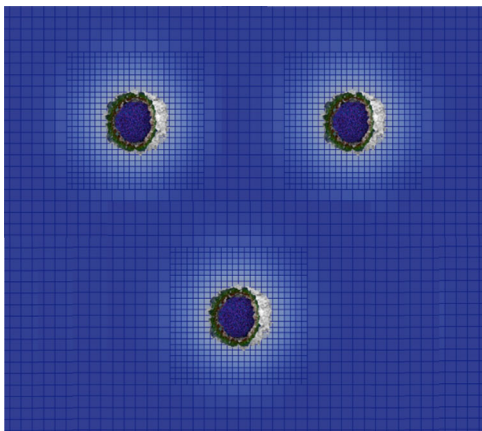
FIG. 17. The flow structure around a macromolecule resolved by a multiresolution mesh being finer in proximity and inside the macromolecule.

This operation implies refining the LB mesh on the fly by following the instantaneous position of the molecular agents. As the macromolecules diffuse and encounter each other, assemble or undergo structural transitions, a finer level of meshes must follow the macromolecule like a shadow; see Fig. 17.

Owing to the simplicity in constructing the Cartesian mesh, the price to pay for automatic mesh refinement is, in principle, negligible although the associated software management can become nonetheless pretty demanding.

The availability of multiresolution techniques makes it increasingly attractive to design fully fledged, LBPD-based multiscale or level environments.

The underlying multiresolution mesh is the basic support to move information between juxtaposed regions, whereas Eulerian and Lagrangian agents cross the boundaries and unveil their essential nature locally in space and time. Note that in this context we refer to Eulerian and Lagrangian as to two distinct levels of description, each of which does embrace multiple scales. Here we follow Noble's definition (Noble, 2016), according to which scale pertains to the physical extent over which information is distributed in space and time, whereas a level touches at the mathematical or computational organization of such information. In this terminology, "transmutation" is basically a change of level.

As a result, at each scale, the following four quadrants are involved (see Fig. 18):

- EE (Eulerian-Eulerian): LB populations that coarsen and refine
- LE (Lagrangian-Eulerian): Particles that transform into LB populations
- EL (Eulerian-Lagrangian): LB populations that transform into particles
- LL (Lagrangian-Lagrangian): Particles that coarsen or refine

As previously discussed, the EE transform can be accomplished within LB-adapted multigrid techniques. The LL transform also involves homogeneous quantities and adjusting their interactions to the required scale appears to be doable within currently existing methods (Praprotnik, Site, and Kremer, 2008). A typical example of LL transform is between
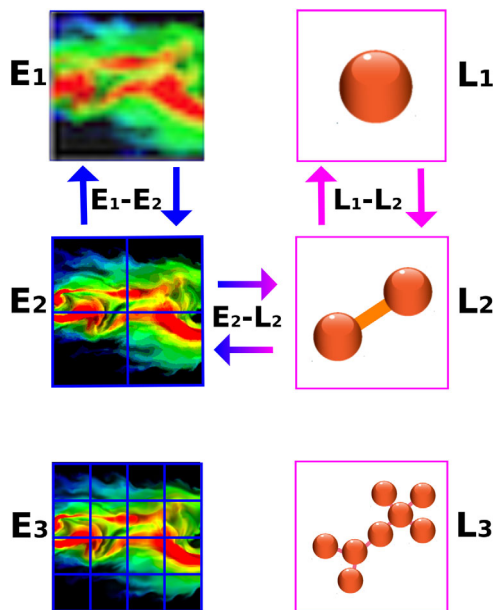

FIG. 18. Diagram for the multiscale approach where Eulerian (E) and Lagrangian (L) agents can move across scales and across representations. The boxes of different size in the left column indicate the coarse or fine grain of the Eulerian representation, and the ball-and-stick representations in the right column indicate particle-based molecules at different resolution. The arrows indicate the exchange of agents between E-E scales, L-L scales, and E-L representations.

a detailed representation of a polypeptide to an elastic network and vice versa. Coarsening implies loss of information by projection and it is easier to handle than the reverse case, namely, reconstruction of the information lost in the projection step, a inherently non-zero-error task. The cross-level terms LE and EL are specific of LBPD, and hence less consolidated.

We note that the mechanism that regulates the crossing is inherently different for LE versus EL transitions, since the single-particle identity is inevitably lost in LE. For the EL case, we need to guess the identity of individual particles, and even more unwieldy, construct the topology of extended molecules.

A fully fledged multiscale or level LBPD framework still awaits for conceptual validation from the point of view of fundamental statistical physics as well as in terms of computational implementations. Hence, making an educated guess on its future is far from being an easy task. Yet we can make a conservative prediction about the concurrent spatiotemporal scales that can be accessed once Exascale computing becomes available.

The LBPD code MUPHY was reported to deliver 20 PetaFlops/s for extreme simulations involving 20×10^9 fluid sites and nearly 70×10^6 particles. By naive linear extrapolation, an Exaflop computer would permit one to scale these figures up by another factor of 50, leading to 1×10^{12} fluid sites and nearly 10×10^9 particles. This corresponds to four decades in space, the best one can expect without any of the multigrid or level sophistications previously described. Once such multiscale or level strategies are in place, another 2 orders of magnitude can reasonably be envisaged (in a plain

multigrid scenario this corresponds to about seven levels of refinement, which is well within the current capabilities of multigrid LB solvers). The resulting LBPD tool would then be able to handle six decades in space and time.

With these advances in mind, several exciting scenarios may open up to the LBPD strategy, described next in some detail.

B. Protein crowding

Protein trafficking in cellular compartments is deeply connected to the structural and diffusional behavior of proteins under crowding conditions. Realizing that the interior of cells is characterized by high concentrations of macromolecules and, depending on the organelle or subcellular location, 5% to 40% of the total volume is occupied by macromolecules, those proteins conduct their activity in extremely crowded environments. Therefore, organelle functioning depends on the structural and dynamical response of proteins to dense packing conditions, a critical yet elusive element of cellular organization. While advances are made through *in vitro* studies, crowding effects can force molecules in cells to behave in radically different ways as compared to test-tube assays (Ellis, 2001; Zhou, Rivas, and Minton, 2008).

The way that cells utilize intracellular spatial features to optimize their signaling characteristics is still not clearly understood. The physical distance between the cell-surface receptor and the gene expression machinery, fast reactions, and slow protein diffusion coefficients are some of the properties that contribute to the intricacy.

Extracellular signals captured by receptor proteins on the cell surface are transduced inward to control target proteins or gene expression. Two interconnected underpinnings of this cellular response are molecular mobility (e.g., diffusion and active transport) and the signal transduction reactions. Despite their importance, limited attention has been paid to the former biophysical properties of the cellular environment, which can contribute to overall signaling characteristics of the system by introducing nonlinear signal delays. The Stokes-Einstein relation implies a slow diffusion rate of protein

macromolecules, which are key players in the signaling. The significance of diffusion in reaction-diffusion systems becomes key whenever reactions are comparatively faster than diffusion rates.

Extremely high protein density in the intracellular space, commonly called molecular crowding, can magnify the spatial effect. In a typical cell, the total macromolecular density is 50–400 mg/ml, far higher than typical *in vitro* conditions (1–10 mg/ml). If a solution contains 30% by volume of identical globular molecules, less than 1% of the remaining space is available to an additional molecule of the same size due to the excluded volume effect caused by steric repulsion, resulting in a mutual impenetrability of macromolecular solutes. In such environment, slow (5–20 times lower than saline solutions) translational diffusion is still observed, which exhibits the footprint of anomalous diffusion (Takahashi, Arjunan, and Tomita, 2005; Sterpone *et al.*, 2014; Timr *et al.*, 2019). Anomalous diffusion is defined as sublinear scaling of mean-squared displacement of the molecule over time and is used as a measure for cytoplasmic crowding. Molecular crowding, as exemplified by two snapshots of simulation shown in Fig. 19, can also alter protein activities and break down classical reaction kinetics.

Ideally, to reproduce crowding effects, with the ensuing anomalous diffusion and protein encounters, simulation methods should be able to track coarse-grained shapes and sizes of molecules and their positions in three-dimensional space. Proteins stay localized in certain compartments as a result of cell compartmentalization and noncovalent weak interactions such as ionic, van der Waals, hydrogen bonds, and hydrophobic-polar interactions. Weak interactions, which can also influence the reaction and diffusion rates of molecules, should be considered during simulation. As the first estimate, the problem being dominated by diffusive motion, one should consider the simulation of a cubic system of side $L = 10^3$ lattice units. For such systems, the LB and PD methods contribute almost equally to the computational effort. With a unit diffusion coefficient in lattice units, the computational complexity scaling as $L^3 T = L^5$ is particularly challenging, resulting in an estimated complexity
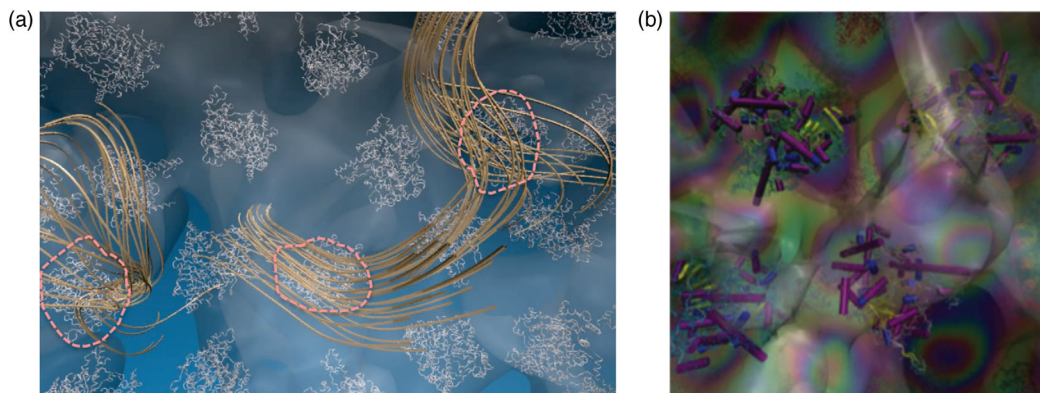

FIG. 19. Snapshot of globular proteins in solution, shown in different visual representations. (a) Proteins are shown with wirelike bundles and the surrounding hydrodynamic signal is represented via an isosurface of its velocity field. Streamlines generated by selected proteins in the crowded macromolecular environment show how the hydrodynamic disturbance propagates in the aqueous solution. (b) Proteins represented via conventional ribbon-sticks style and the constant velocity isosurface that illustrates the complexity of the flow structure.

$$\mathcal{C} \sim 10^1 - 10^2 \text{ ExaFlops.}$$

Clearly, high scalability is mandatory to support the handling of such large intracellular systems.

Biological interactions are quite promiscuous and their occurrence cannot be predicted based on conformational considerations. It is essential to account for chemical specificity in order to capture the local details of molecular recognition and signaling. LBPD stands as an appealing approach to address the plethora of questions related to signaling in the protein crowd.

Having drastically reduced the number of degrees of freedom for the water solvent down to the stringent economics of the LB method, the PD engine should already be sufficient to track enzymatic activity and related pathways. This is sufficient for a subset of possible conditions in the cell, but still not satisfactory for the general case. As described earlier, an essential feature is to resolve the local hydrodynamic, thermodynamic, and chemical patterns in the surroundings of active sites, with their peptidic components, hydrogen exchanging groups, or metal atoms. Local mesh refinement entails dramatic benefits in terms of reducing the cost of the far-field regions, and so would the corresponding adaptive time stepping. Leveraging the minimalistic LB mesh is precisely the option at stake to reduce the number of degrees of freedom (by a factor of 2^3 gain) together with the time step (by a factor of 2 gain) at every resolution change. The same advantage applies for every partial differential equation that can be solved on the fly via a concurrent LB scheme, say the ADR equation, for electrostatics, or for other modeling purposes. If the same LB mesh is used for solving concurrent PDE via conventional finite-difference approaches, one should refer to the consolidated multigrid algorithms.

C. Direct simulation of full-scale cell compartments: Golgi and neuronal firing

Cellular organization relies upon the operation of several organelles, from the nucleus to ribosomes, mitochondria, lysisomes, Golgi, etc. Each organelle lives its own life separated by confining lipid bilayers. These specialized subunits are in action and continuously interplay with one another. The scenario is extraordinarily rich and no numerical framework can hope to capture the entire picture. Yet a few characterizing elements make the LB framework particularly appealing. With the Exascale capability at our doorsteps, the LB method can be taken to the next level: from the study of basic biofluidic processes to the direct simulation of full-scale cellular compartments, such as protein cargoes, vesicles, and possibly even full-scale organelles. The endeavor commands the integration of the LBPD paradigm within broad scope software infrastructures, including mechanical models of biological structures, ranging from all-atom molecules to elastic networks for membranes and so forth (Chiricotto, Sterpone et al., 2016).

In order to design the most appropriate strategy, it is instructive to take a look at a few processes involving macromolecules that are transported within the cellular environment. During the cell life cycle, proteins are continuously translated and delivered to specific cellular locations by traversing different membrane structures. However, most molecules, including proteins, are too large to pass directly through membranes. Instead, large molecules are loaded into small membrane-wrapped containers called vesicles. Vesicles are constantly forming—especially at the plasma membrane, the endoplasmic reticulum, and the Golgi apparatus, or simply the Golgi. Once formed by exocytosis, vesicles deliver their contents to destinations within or outside the cell. On the different, yet related, scenario of neurotransmission, signaling neurotransmitters are released by a neuron and bind to and activate the receptors of another neuron. Neurotransmission is the essential process of communication between two neurons with synaptic transmission and firing relying on the release of neurotransmitters. The latter are stored in vesicles in the axon terminal. Different mechanisms involve partial opening and then reclosing of vesicles, together with the fusion of vesicles with the membrane.

The emerging picture is that the entire arsenal of the LBPD approach, entailing single or multiphase variants, in presence or absence of suspended macromolecules, provides a powerful, flexible and self-contained framework to solve multiple levels of cellular biology. A closer look at how proteins and vesicles interplay unveils the type of challenge the numerical approach has to tackle. The way proteins are transferred inside in cellular compartments is fascinating. Vesicles form when the membrane bulges out and pinches off. Then it travels to its destination, where it merges with another membrane to release its cargo. In this way, proteins and other large molecular cargoes are transported without ever having to cross a membrane. Even more, the mechanism underlying the formation of vesicles is budding and is deeply assisted by proteins. When vesicles bud, they wear "coats" and when coat proteins assemble at the member, they force the lipid bilayer to begin to bend. As they gather at the membrane, coat proteins may also select the specific cargo that is packaged into the forming vesicle. As more coat proteins are aged, they shape the surrounding membrane into a sphere. Finally, once a coated vesicle pinches off, the coat falls off, and the cargo-filled vesicle is ready to travel to its final destination. The plain fact shows that the proteins-fluid system is inherently two way such that chemical specificity as much as the vesicular chemical composition has to be correctly included to convey the required molecular realism.

The feasibility of reproducing full-size organelles can be analyzed by considering the typical size of the Golgi. This organelle has a lateral size of 2.5 μm and is composed of 10^{12} atoms, that is, the "TeraSize." The typical time span for morphogenesis spans between 1 μs and a minute. In operating conditions, one femtosecond time step is required for the accurate, bottom-up description of molecular trajectories. Given the burgeoning progress of computing power promises for the forthcoming years to reach the ExaFlops/s capabilities. Sustained by the development of modeling techniques and specialized algorithms, the available power will soon allow facing astonishing assemblies of macromolecules above the microsecond time scale and, at the next level, targeting entire cellular compartments. Under such conditions, one can expect that, on an Exascale computer, one could simulate the system evolution in full at a cost of
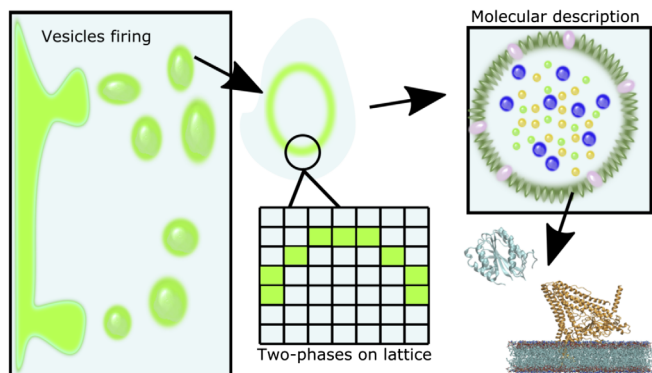
FIG. 20. Multiscale representation of vesicles transporting proteins. Vesicle firing from a membrane where the vesicles are modeled as immiscible fluid phase separating from the aqueous host. The central panel zooms the vesicle bilayer with liquid water in the interior. The right panel depicts a molecular representation of the vesicle containers with an heterogeneous protein suspension in the interior and membrane proteins embedded in the bilayer.

$$\mathcal{C} \sim 10^4 \text{--} 10^6 \text{ ExaFlops}$$

corresponding to about 1–10 days of wall-clock time.

Clearly, the path to simulate full-scale compartments can be a multistage process and the first step is to represent lipidic membranes at a coarse LB level, namely, by neglecting the molecular character and chemical specificity of the confining components, as sketched in Fig. 20. Being a peculiar visco-elastic fluid, the membrane can be handled by the multiphase LB, e.g., via Shan-Chen or free-energy approaches. The description has to cope with the fact that a membrane has finite thickness; therefore any multiphase approach should reproduce not only the interfacial properties of the lipidic bilayer, but also its phase diagram, including the multiple shapes of micelles, vesicles, etc., with their tendency to deform. The formation of lipidic chains and their preferential orientation under external forces induce changes in shapes from circular to elongated, as observed in experiments. Complex fluid-fluid interfaces featuring mesoscale structures with adsorbed particles are key elements of biological components. For such a schematic approach, the expectation is demanding but one should keep in mind that confiners and carriers can provide a minimal, yet satisfactory, level of realism.

D. Biochemical reactivity and signaling pathways

As discussed earlier, LBPD can reproduce the interplay between flow and macromolecules in a relatively large, cell-like environment. Dealing with biochemical reactivity is a different, large sector that calls for a full deployment of the simulation capabilities (Takahashi, Arjunan, and Tomita, 2005). Application of the LB method to reproduce advection-diffusion-reaction phenomena is an option. However, biochemical reactions are not always classifiable according to simple statistical rules, as for the Michaelis-Menten or logistic laws (Voit, 2013).

In biology details matter since a small minority of active sites immersed in a jungle of organic groups can make the whole difference. In addition, metabolic and synthetic reactions can occur in bulk conditions within a single phase (homogeneous reactions) or in a multiphase environment, typically at the interface between regions (heterogeneous reactions), in no-flow or flow conditions, as, for example, the enzyme reactions on the surface of the blood vessels.

Although there are many possible reactive events in action, they all fall into two broad categories: oxidation and reduction, the motion of functional groups within or between molecules, the addition and removal of water, and the bond-breaking reactions. Most reactions are catalyzed by proteins, RNA, or DNA. A different class of reactions involves electrostatic, electrodynamic, and hydrophobic interactions, where electrons are not shared and covalent bonds are not modified. Enzymes have the property to increase the rate of the reactions and are specific to the reactant molecules, also known as the substrates, interacting with high affinity. Finally, the activity of enzymes is regulated in a number of ways, controlling the rate and amount of products formed. Examples of regulation include cofactors binding to the enzymes or the presence of reaction products that inhibit the reaction (Fersht, 2017).

The range of biological activity does not result from many different types of reactions, but rather from a few simple reactions, occurring under many different situations. Thus, for example, water can be added to a carbon-carbon double bond as a step in the breakdown of many different compounds, including sugars, lipids, and amino acids. One could model this compact set of reactions by utilizing the LBPD apparatus. Within a classical description this is feasible indeed, except that bond breaking and formation requires using electronic structure methodologies, notably by using one of the most successful theories to date, the quantum density functional theory. The good news is that, given the small number of active sites present in macromolecules, this stands as a perfect candidate to embed a numerical solver for electrons within a classical solver for particles, and ultimately within the LB overarching framework. The quantum-mechanical molecular mechanics (QM-MM) approach is an active avenue of research today and proceeds along similar ideas as the LBPD scheme. Clearly, the numerical details of the LB and QM methods are different in nature, but basic similarities can be found. Indeed, like LBPD, the QM-MM procedure is based on a combination of Lagrangian (classical molecular dynamics) and Eulerian (quantum electronic structure) components. One may push the similarities even further by utilizing the LB method to solve the electronic structure too. Work in this direction (Mendoza, Succi, and Herrmann, 2014) is promising for the future.

To the point that, in the far distance, one may envisage an unprecedented four-level QM-MM-PD-LB unified multiscale structure that ranges from electronic scales all the way up to the cellular ones, the overlap link being MM-PD. Incidentally, three-level structures of this sort have now been in place for two decades, although their routine operation seems to remain somewhat unwieldy (Abraham *et al.*, 1998).

In prospect, evolutions of the LBPD methodology to cope with reacting systems should benefit from one of its major

assets. Reactions control species interconversion and involve the breaking and forming of covalent bonds, often catalyzed by enzymes. Dealing with particles or molecules that change identity on the fly, their molecular connectivity and can undergo multiple reaction channels, is a tall order for the Lagrangian treatment. Instead, electron transfer and chemical breaking and forming can take advantage of the probabilistic nature of the kinetic representation. In this respect, one can envisage interconversion between the Eulerian and Lagrangian representations for molecules that are likely to react, and choose between the optimal treatment, possibly on the fly. An estimate of the computational resources must embrace the full-scale organelles, for which a region of $L = 10^4$ sites in size must be catered to. With diffusive-reactive scaling, $T \simeq L^2$ and some $k \simeq 10^5$ flops/(site/step), we obtain

$$\mathcal{C} \sim 10^7 \text{ ExaFlops}$$

corresponding to about three months wall-clock time on a Exaflop computer.

Assuming sufficient accuracy to model the chemical reactions is available, the next objective is to examine the structure and dynamics of cellular functioning at the system level, rather than the characteristics of isolated regions. Collective properties of networks, such as their efficiency and robustness, emerge as a central characteristic in system biology. Needless to say, the understanding of these properties may deeply impact medicine, bolstering the emerging notion of network medicine, as opposed to the "magic-bullet" approach of genomics. Advances in this direction can take full advantage of LBPD once sufficient chemical specificity is incorporated via coarse-grained force fields. Caution must be exercised to find the optimal balance between resolving local details and the collective properties of the metabolic network. As it stands, this is an ideal scenario to develop innovative multiscale or level methodologies for the biological context.

E. Hemostasis

Another grand challenge is the study of hemostasis, a crucial healing mechanism in which molecular specificity and chemical reactivity contribute on an equal footing. Hemostasis is the immediate response of the body to stop bleeding from within a damaged blood vessel. It is the first stage of wound healing and involves a blood change from the liquid to the gel state (coagulation). When an endothelial injury occurs, the endothelial cells stop secreting coagulation and aggregation inhibitors and secrete instead the globular glycoprotein von Willebrand factor (vWF), which uncoils and initiates the maintenance of hemostasis after injury. The overall process is governed by Virchow's triad which comprises composition of blood, wall surface reactivity, and material flow. A multiscale simulation approach stands out as a prime route to gain a better understanding of such a complex and life-essential mechanism.

As we delve into the details, it becomes clear that hemostasis shows articulated features, as sketched in Fig. 21, highlighting the multistep and multiscale elements in action. It proceeds along three subsequent steps that seal the injury until tissues are repaired: vasoconstriction, temporary blockage by a platelet plug, and the formation of a blood clot. Vascular spasm is the first response to constrict the blood vessels and reduce the blood loss. Second, platelets stick together to form a temporary seal via the so-called primary hemostasis: platelets adhere to damaged endothelium to form the plug and then degranulate as activated by the vWF. Finally, coagulation takes place and reinforces the platelet plug with fibrin threads that act as the "molecular glue." In this picture, platelets are key to the process: the plug forms almost directly after the vessel has ruptured and within seconds and disrupted platelets adhere to the subendothelium surface. Within a minute the first fibrin strands begin to intersperse among the wound and just a few minutes later, the plug is completely formed by fibrin. During the process, a dozen clotting proteins are activated in a sequence known as the coagulation cascade
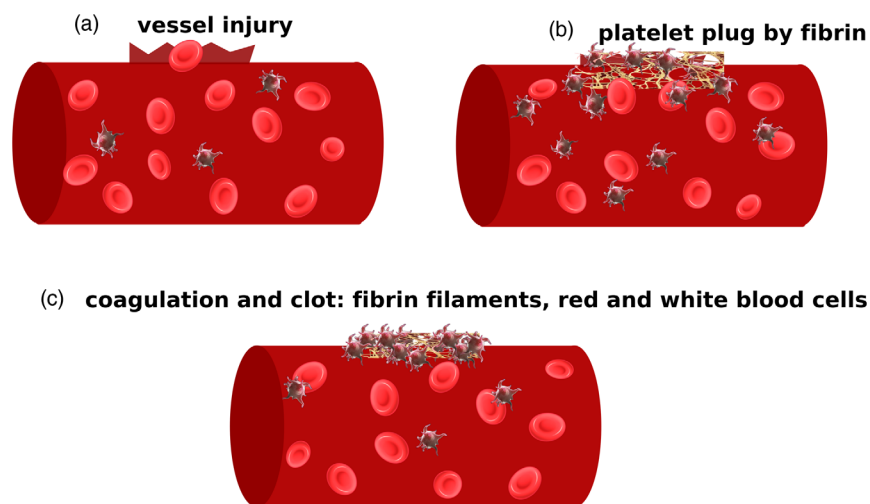

FIG. 21. The major phases of hemostasis following the vessel injury: (a) a platelet plug is formed to rapidly stop the initial bleeding; (b) a mesh of fibrin is made at the injury point to make the platelet plug stronger; and (c) finally the clot is formed by the coagulation of fibrin, red and white blood cells.

to hold it in place, the so-called secondary hemostasis. Here red and white blood cells are trapped in the mesh which causes the primary hemostasis plug to become harder: the resultant plug is called a thrombus or clot.

The role of blood flow is even more important than it might appear at first sight. Since the 1970s, experiments have demonstrated that shear stress strongly affects the activation of platelets and their adhesion to the injured tissue. However, there is still confusion about the mechanism that regulates platelet arrangement and about the reasons why sites of disturbed flow appear to be more prone to platelet deposition.

When the level of shear exceeds a certain threshold, platelets aggregate even in the absence of any chemical agonist and without any modification of the vWf. The reason for this behavior remains unclear. Shear-enhanced exposure or the alteration in the structure of receptors on the platelets membrane increases the frequency of collisions where particle migration is known to increase with shear up to 3 orders of magnitude above the Brownian value, due to the enhanced collision frequency, primarily between platelets and erythrocytes. Furthermore, erythrocytes are known to enhance shear-induced platelet adherence not only mechanically, but also chemically, through the release of the platelet agonist adenosine diphosphate.

The uncoiling of bound globular vWf at elevated shear is responsible for the increased platelet deposition and upon platelet activation, the release reaction feedback amplifies the hemostatic system. At high shear, adhesion requires the synergistic action of several receptors on the platelets membrane and on other ligands. Upon activation, platelets change their shape from discoid to spherical, release their granule content, and increase their stickiness among themselves, as a function of the amount of local shear stress. Chemistry plays a major role, as a multitude of reactions lead to the production of thrombin, which is a key enzyme in the hemostatic process as well as a strong platelet agonist. In the course of hemostasis, fibrin fibers which surround the platelet aggregate and stabilize it against the shear forces in the flowing blood.

Shear stress and saturation-dependent changes in surface reactivity influence thrombus growth and the adhesion and aggregation of platelets to reactive materials. Simplified models based on fluid dynamic and species conservation equations can match *in vitro* experimental data, as regarding the initial phase of platelet deposition, when thrombus growth can be neglected, while accounting for shear stress and changes in surface reactivity. However, taking thrombus growth into account results in a free-boundary problem, with fully coupled fluid dynamic and species conservation equations, again a scenario that calls for the LBPD apparatus, as witnessed by a few pioneering studies (Ouared and Chopard, 2005; Chopard, Ouared, and Rüfenacht, 2006; Harrison *et al.*, 2007, 2008; Ouared *et al.*, 2008; Tamagawa *et al.*, 2009). In these studies, activation of platelets in the bulk flow and subsequent agonist production were not included as a part of the model. On the other hand, flow could be explained by using a shear-independent adhesion rate. By using such a model, predictions on the flow structure were improved in some parts of the flow chamber, such as in stagnation points, whereas notable discrepancies remained in some other parts.

In order to be effective, the LBPD approach should include the combined effects of shear stress, changes in surface reactivity, and aggregate growth in modeling both hemostasis and thrombosis. The practical outcome would be paramount, i.e., assist the minimization of thrombus formation in vascular prostheses without the use of strong anticoagulants. This is all important in bioengineering to design materials with improved surface properties, also for shape optimization techniques in flow conditions versus platelet deposition. Because of its fundamental character, the influence of shear stress demands the inclusion of the full coupling of flow and thrombus growth in models intended to capture the long-term behavior of platelet deposition, with the potential to enlighten the basic mechanisms taking place in further kinds of adhesion processes. Given the extended multiscale nature of hemostasis, Exascale computers may still be insufficient to solve the problem. For an estimate of computational requirements, we take a cubic box of side $L = 10^4$, in order to acknowledge the need for micron-scale resolution of regions of the order of the centimeters. With a diffusive-reactive scaling $T \simeq L^2$ and a computational density of the order of $k \simeq 10^6$ flops, leading to

$$\mathcal{C} \sim 10^8 \text{ ExaFlops.}$$

Such a value could be reduced by making systematic assumptions on the process and analyzing the different phases of primary and secondary hemostasis at different stages.

Notwithstanding the mentioned current limitations, it is of paramount importance to consider the simulation of hemostasis for medical purposes. Hemostasis is life essential because it can go wrong in atherosclerotically narrowed vessels. Ruptured plaques and elevated shear rates may induce the formation of platelet-rich thrombi that may eventually become life threatening by occluding the vascular lumen. In fact, a major number of deaths is due to thrombotic events provoked by disorders of the hemostatic system. Severe consequences are triggered if the thrombus detaches from the vessel wall and travels through the circulatory system. If the clot reaches the brain, heart, or lungs, it can lead to a stroke, heart attack, or pulmonary embolism, respectively. Here again the full potential of LBPD for medical purposes cannot be understated, and a full-scale deployment of the method in complex arterial networks should be considered. As before the large spread of scales, the presence of multiple agents, and the need for specialized solvers for chemical reactivity calls for similar, if not more sophisticated, high-performance techniques that push the computational limits to their extreme.

VIII. PCB MODELING VERSUS BIG DATA SCIENCE

All throughout this review, we have advocated the mesoscale physics-inspired modeling of complex phenomena at the interface between physics, chemistry, and biology as a promising avenue toward the ultimate goal of benefitting medical science and clinical practice. Before concluding, it is worth mentioning prospective connections of LBPD with the current trend toward the use of large data and machine-learning techniques in science. Putting aside the most aggressive instances of large data (Anderson, 2008), which can be

readily commented away (Coveney, Dougherty, and Highfield, 2016; Hosni and Vulpiani, 2018; Succi and Coveney, 2018), it is undeniable that data science, and notably physics-aware machine learning (PAML), bears major potential to enhance the LBPD scenario. By PAML we refer to the machine-learning scenario whereby neural networks are designed in such a way as to incorporate physical constraints directly into their architecture; see Karpatne *et al.* (2017) and Raissi and Karniadakis (2018) and references therein.

We note in fact that, due to its inherent mesoscale nature, LBPD will necessarily be exposed to increasing parametrizations, as it proceeds toward enhanced biological fidelity. For instance, PAML techniques could prove of great value in automating the search of effective potentials providing an optimal match to the desired biological properties, such as the mechanical response of red blood cells (size, shape, stiffness, etc.), so as to improve the description of their interaction with tissue cells. More ambitiously, PAML could even help in automating the choice of the relevant degrees of freedom which characterize the mesoscale formulation of the problem, thus helping to strike an optimal balance between computational efficiency and biological fidelity.

IX. SUMMARY AND PERSPECTIVE

The lattice Boltzmann method has undergone major progress over the last decade, moving from an alternative technique for solving Navier-Stokes hydrodynamics to a versatile computational strategy to simulate complex states of matter across many scales of motion, including microflows and nanoflows of relevance to biological processes. This quantum leap has been fueled by major advances of the LB "technology" alone and by its successful coupling to particle methods, i.e., the lattice Boltzmann–particle dynamics paradigm illustrated in this review.

The lattice Boltzmann–particle dynamics paradigm has given access to a new level of complexity in the description of phenomena occurring at the physics-chemistry-biology interface. In this review, we have focused our attention on the possibility of reaching up to scales of direct relevance to clinical applications, thus portraying the grand dream of a mesoscale physics-based approach to precision medicine. This grand dream has been illustrated through a series of actual examples which, albeit not quite there yet, support the expectation that, once Exascale computing is with us, the dream will come true. The task is neither simple nor straightforward, but its scientific and societal impact cannot be overstated.

APPENDIX: HIGH-PERFORMANCE LBPD COMPUTING

The LBPD paradigm described in the present review connects two basic computational pillars, a lattice-bound treatment of the fluid field with an off-lattice handling of the discrete particle dynamics. Given the markedly distinct nature of the associated data structures, the optimal merge of these two components must necessarily be realized through a careful trade-off between the two. In this Appendix we provide an overview of the main technical issues which concur to achieve such compromise.

1. High performance simulations of particle dynamics

The computational requirements of PD simulations have always limited their applicability to short time intervals, but advances in parallel algorithms and special-purpose hardware (GPU, FPGA, ASIC, etc.) have recently extended the scope of such simulations to much longer time scales. The state-of-the-art platform for high performance and parallel execution of PD simulations is the Anton 2 system developed by DE Shaw Research (Shaw *et al.*, 2014). Anton 2 performs the entire PD computation within custom ASICs that are tightly interconnected by a specialized high-performance network. A key component of the Anton 2 design is a set of new mechanisms devoted to efficient fine-grained operations. The resulting architecture exploits at its best the parallelism of PD simulations, which fundamentally consists of a large number of fine-grained computations involving individual particles or small groups of them. By providing direct hardware support for fine-grained communication and synchronization, Anton 2 allows these computations to be distributed across an increased number of functional units while maintaining high utilization of the underlying hardware resources. Fine-grained operation is exposed to software via distributed shared memory and an event-driven programming model, with hardware support for scheduling and dispatching small computational tasks. Anton 2 breaks the microsecond-per-day barrier on million-atom systems, allowing larger biomolecules such as ribosomes to be simulated for much longer time scales.

2. Achieving high performance for lattice Boltzmann methods

In general the performance of the LB method on most platforms is memory bandwidth limited, that is, the rate by which the set of LB populations can be read off and written to the memory is the main bottleneck. This issue and its consequences can be understood looking at, for instance, the widely used D3Q19 model in which there are 19 populations that need to be read and written twice from the memory. Data are moved from or to memory once for the collision phase and once for the streaming phase. If the size of the memory word used to store each population is WS, then $4 \times 19 \times WS$ bytes are moved for each point of the lattice. The number of floating point operations depends on the collision operator only (there are no floating point operations during the streaming phase) but it is safe to assume that it does not exceed 300 so, using single-precision floating point

format ($WS = 4$), the arithmetic intensity[1] of the LB update procedure is $300/4 \times 19 \times 4 \simeq 1$. On virtually any modern platform the number of bytes that can be moved from or to the memory in a unit of time (e.g., in a nanosecond) is smaller that the number of floating operations that the computing cores can execute during the same unit of time (assuming that the operands are available in the registers) so it is the memory that limits the throughput also in the ideal situation in which the access to the memory achieves its peak performance. The performance of LB codes is typically given in terms of millions of fluid nodes updates per second (MFLUPS). On a platform with a memory bandwidth B_{max}, the peak performance in MFLUPS is

$$P_{MFLUPS} = \frac{B_{max}}{4 \times 19 \times WS \times 10^6}.$$

However, to alleviate the problem of the memory bandwidth several variants of the LB method have been proposed. One of the most widely used is the so-called fused implementation, in which the collision and the streaming phases of the LB update are combined in a single procedure that either reads the populations from the source locations and collides them (prestream) or collides them and stores the result directly to the target locations (poststream). The advantage is that the populations are read from and written to the memory just once for each time step.

In this way the achievable peak performance doubles. A fused implementation increases the complexity of the code (and, most of the times, the memory requirements). Unfortunately, the actual performance may be significantly lower than P_{MFLUPS} since the memory bandwidth is hardly exploited at its best due to the memory access pattern of the LB method. The situation is the same for any platform (i.e., a general purpose CPU or an accelerator) but the techniques to improve the situation, that is to reach a higher percentage of P_{MFLUPS}, strongly depend on the features of the memory hierarchy and on the programming model. For instance, there are several alternatives for the data layout of the populations in memory. It is possible to store all populations of a single lattice point close to each other in a data structure containing 19 floating point values. The populations of all lattice points will form an array of structures (Fig. 22) that can exploit the data-locality principle of cache-based memory hierarchies like those found in a general purpose CPU. However that layout prevents one from exploiting vector instructions available, for instance, on Intel CPU, because populations of different mesh nodes are stored in noncontiguous memory locations. For this reason, especially on accelerators like the Nvidia GPUs, a different layout, in which each population is stored in a single array and the whole data set of populations forms a structure of arrays, must be used so that neighboring threads access contiguous memory locations according to a principle of thread locality. Many studies have been carried out, some of them in the recent past, proposing variants in the usage of
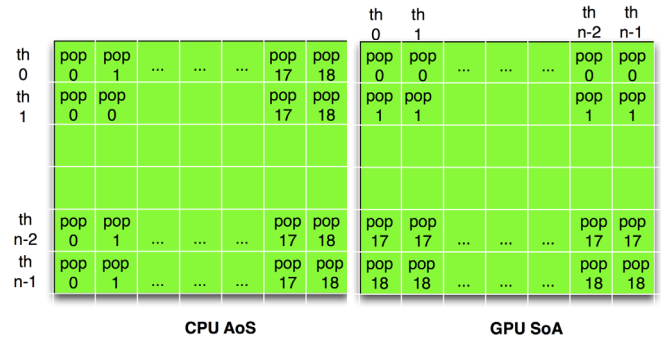

FIG. 22. CPU vs GPU optimal data layout. Row major ordering is assumed for storing multidimensional arrays in memory (typical of *C* and C++ languages). AoS stands for array of structures; SoA stands for structure of arrays.

these two data layouts with special attention to the Nvidia GPUs which expose a complex memory hierarchy to the explicit control of the programmer. Although their main purpose is the optimization of the LB update, those works provide useful indications to enhance the performance of other procedures, like those for the solution of PDE, that need to use 3D *stencils* to access data.

Recently, new technologies promise a significant boost in performance for any memory bandwidth-limited application, including the LB method. In particular, multidimensional memory-processor interfaces provide a much higher bandwidth. Preliminary tests with a recent generation Nvidia GPU card (featuring the so-called Pascal GPU architecture) that offers a memory bandwidth up to 720 Gbytes/s show an improvement of the LB performance of a factor of 3 with no change in the source code.

As mentioned several times in this review, the populations update procedure of the lattice Boltzmann method is suitable to parallel processing and usually achieves good efficiency on shared memory systems. The only drawback is that a parallel fused implementation requires a double memory buffer for storing the populations: at each iteration, one of the buffers is used as a source of the populations and the other as a target; at the end of the iteration the role of the two buffers swaps. Actually, following a tricky ordering, the collision and streaming phases could be carried out without requiring a double buffer but only with a serial update procedure. As a consequence, a parallel implementation of the fused procedure for the D3Q19 model requires, at least, $19 \times 2 \times WS$ bytes of memory for each lattice site (actually more, because memory is required also for the hydrodynamic variables, i.e., density, velocities, etc.) so that large-scale simulations may not fit in the memory of a shared memory system. In those cases or simply for reducing the simulation times by exploiting many more computing resources, it is necessary to resort to a distributed system with multiple computing nodes. We discuss the main issue of that approach in Sec. A.3.

3. Overlap between computation and communication

Exascale computing platforms will very likely be based on super clusters of powerful computing nodes, possibly equipped with accelerators like GPUs. A general and detailed

---

[1]The arithmetic intensity of a numerical procedure is defined as the ratio between the number of floating point operations and the number of bytes moved from or to the memory.

discussion of the challenges posed by the efficient exploitation of such platforms is beyond the scope of the present work; however, at least one specific issue, related to the scalability of large-scale LBPD simulations, deserves to be mentioned, namely, the need of overlapping the computation and the communication stages of the LB algorithm, so as to "hide" the overhead of the latter behind the former.

When more than one computing node (CN) is available for the simulation of a system, it is quite natural to apply a *domain* decomposition. With this approach, each computing node is responsible for a subset of the whole mesh. Here we assume that the domain has a regular geometry postponing the discussion of irregular and/or sparse geometries to Sec. A.4. When N_{CN} are available, each CN i needs to know for the update of the nodes in its own boundaries, the value of variables belonging to the nodes in the boundaries of its neighbors, so, at each iteration, it must (assuming a simple one-dimensional domain decomposition and periodic boundary conditions)

(1) Send data belonging to the points of its bottom boundary to CN $(i-1)\%N_{CN}$.
Send data belonging to the points of its top boundary to CN $(i+1)\%N_{CN}$.
(2) Receive data sent by CN $(i-1)\%N_{CN}$.
Receive data sent by CN $(i+1)\%N_{CN}$.
(3) Update data belonging to the nodes of its subdomain i (both bulk and boundaries).

That "naive" scheme is represented in Fig. 23(a). We define it as naive because computation and communication are carried out one after the other whereas they can be overlapped to a large extent when accelerators are used for the computation. We now briefly describe how the overlap works for one of the most common accelerators in use at the present time.

a. Effective multi-GPU CUDA programming

CUDA, the programming environment of the Nvidia GPU, supports concurrency within an application through *streams* (Cuda C, 2019). A stream is a sequence of commands that execute in order. Different streams, on the other hand, may execute their commands out of order with respect to each other or concurrently. By using two streams on each GPU it is possible to implement the following scheme that assigns one stream to the bulk and one to the boundaries of the LB domain:

(1) Starts to update the boundaries by using the first stream.
(2) First stream:
   • copy data in the boundaries from the GPU to the CPU;
   • exchange data between nodes by using message passing interface (MPI);
   • copy data in the boundaries from the CPU to the GPU.
(3) Second stream:
   • updates the bulk.
(4) Starts a new iteration.

The overlap with this scheme, also shown in Fig. 23(b), is between the exchange of data within the *boundaries* (carried out by the first stream and the CPU) and the update of the bulk (carried out by the second stream). The CPU acts as a data-exchange coprocessor of the GPU. Nonblocking MPI
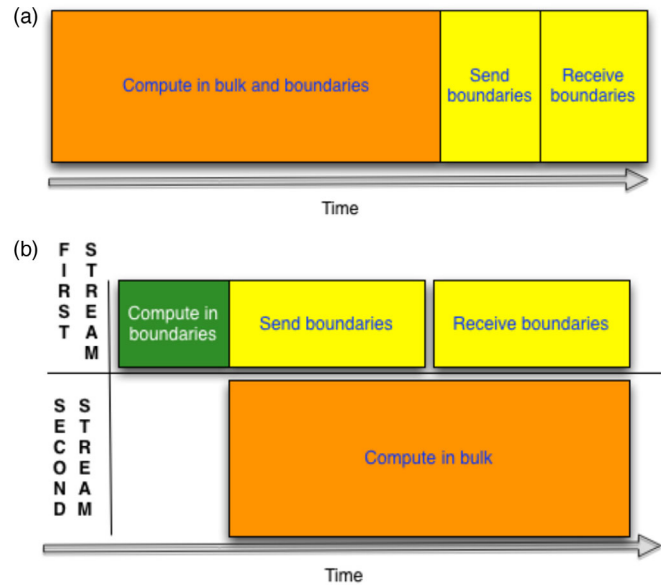

FIG. 23. Communication schemes: (a) with no overlap between communication and computation and (b) multi-GPU scheme using two streams.

primitives should be used if multiple CPUs are involved in the data exchange.

Recently, Nvidia announced NVLink, a new high-speed interconnect technology for GPU-accelerated computing. Supported on SXM-2 based Tesla V100 accelerator boards, NVLink significantly increases performance both for GPU-to-GPU communications and for GPU access to system memory. Programs running on NVLink-connected GPUs can execute directly on data in the memory of another GPU as well as on local memory. That feature should further improve the scalability of LBPD simulations running on large clusters of GPUs.

4. Sparse and irregular geometries

A number of LB applications may use regular and dense geometries for which domain decomposition is, most of the time, straightforward (e.g., a uniform decomposition along one, two, or three directions). However, in biofluidics, soft matter, or porous media simulations the geometry is often neither regular nor dense. In those situations it is not possible or, at least, it is a waste of memory to store LB populations in a simple, regular multidimensional data structure whose size would be proportional to the *bounding box* of the domain. It is much more convenient to follow other approaches for storing only the minimal set of populations required for the simulation of nonsolid nodes of the mesh. In the present section we describe two possible alternatives.

a. Indirect addressing

The first solution relies on a linearized indirect addressing scheme (Dupuis and Chopard, 1999; Schulz *et al.*, 2002). Each node of the LB lattice is labeled with a tag that identifies it as belonging to a specific subregion of the computational domain (i.e., fluid, wall, inlet, outlet, or solid). Mesh nodes
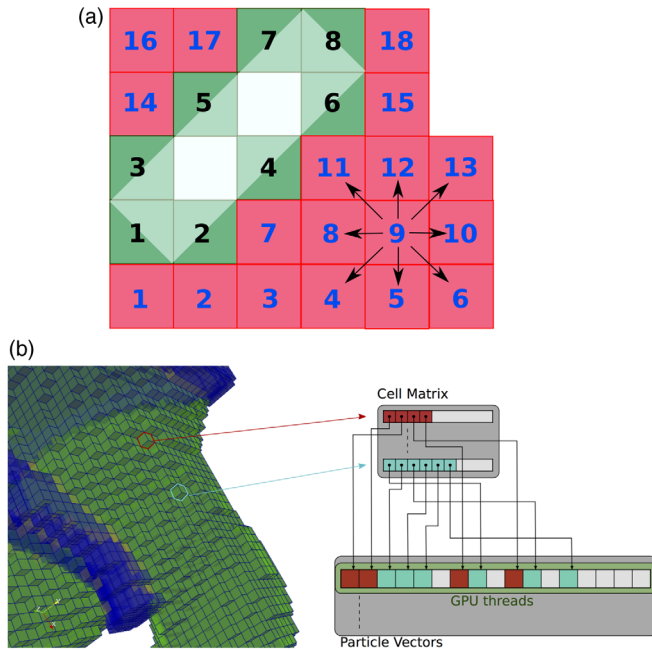
FIG. 24. Organization of mesh nodes according to the indirect addressing scheme for an irregular domain. (a) In the D2Q9 scheme, red (dark gray) squares correspond to fluid regions and green (light gray) squares to wall regions. Inactive nodes are in white and are not numbered since they are not stored in memory. Entries in the connectivity matrix are shown for fluid node 9. (b) Indirect addressing on the GPU for a 3D case. Each GPU thread handles a subset of fluid nodes and a subset of populations.

may be grouped according to their features in several one-dimensional arrays, so that there is an array of fluid nodes, an array of wall nodes, an array of inlet nodes, etc., with the exception of solid nodes that do not need to be stored at all since they refer to inactive regions of the domain.

As a consequence, homogeneous nodes (i.e., all fluid nodes, all wall nodes, etc.) are contiguous in memory regardless of their geometrical distance. This type of data organization requires, for each node, an additional data structure (*connectivity matrix*) that contains the list of all positions, within the previous mentioned one-dimensional arrays, of its neighboring nodes; see Fig. 24 for a simple 2DQ9 case.

With this approach only the nodes playing an active role in the lattice Boltzmann dynamics need to be accessed and stored in memory, resulting in a large savings in storage requirements, despite the additional data structure (Axner et al., 2008), for most nontrivial geometries. An indirect addressing scheme allows one to support very flexible domain decomposition strategies, a fundamental requirement for a good load balancing among computational tasks. For instance, the MUPHY code (Bernaschi et al., 2010) supports all possible Cartesian decompositions (along X, Y, Z, XY, XZ, YZ, XYZ) and *custom* decompositions, e.g., those produced by graph or mesh partitioning tools like METIS(MET) or SCOTCH(SCO), which are necessary for distributing the computational load in an even manner in the case of very irregular domains.

Those graph-based procedures utilize, most of the time, a graph bisection algorithm that is completely unaware of the

geometry of the computational domain. However, the lack of geometrical information degrades the quality of the partitioning as the number of partitions increases, in which case the subdomains reduce to highly irregular shapes with large contact areas between subdomains and large communication loads. A possible solution is to combine the graph-based partitioning with a flooding-based approach (also known as the graph-growing method) according to the following procedure: the mesh is first partitioned, in a given number of subdomains (e.g., 256). If the mesh needs to be partitioned in a finer number of parts, say $256 * P$, with P an integer ≥ 2, then each of the 256 domains is further divided according to the following flooding scheme: starting from a seed mesh point, a region is iteratively grown in an isotropic way until the number of mesh points equals N_i/P (with N_i being the number of mesh point in the ith original partition). As the condition is met, the visited mesh points are assigned to a computational resource. Subsequently, a new growth procedure starts from a new seed until all points in the subdomain are assigned to a new computational resource. Bernaschi et al. (2011) showed how the distribution of tasks versus the number of neighbor tasks with which they exchange data tends, for a large number of tasks, to stabilize instead of increasing up to much higher values as would happen with a pure graph-based partitioning approach.

b. Tiling and blocking

Another possible solution is to "tile" the sparse geometry using much smaller (with respect to the original domain size) regular, square, or cubic (depending on the dimension of the original domain) tiles (Tomczak and Szafran, 2017). One of the advantages of the tiling is that during a single LB iteration, the tiles can be processed independently and in any order provided that values at the tile edges are correctly propagated. Moreover, the update of each square or cubic tile can be carried out according to the simple addressing scheme used for the case of regular geometries. However, a tiling procedure also introduces some overheads due to the presence of solid nodes inside tiles and additional memory requirements for saving information about tiles placement. An interesting variant of this approach has been described by Feichtinger et al. (2011) where a given geometry is divided using a hierarchical structure of "patches" composed of "blocks." For sparse geometries, empty blocks can be removed reducing memory usage and computational complexity. Blocks correspond to leaves in a distributed forest of octrees[2] and are quite sophisticated data structures designed rather for efficient multiprocessor implementations, where load balancing and communication may affect performance. This approach quite naturally supports grid refinement procedures; however, the load balancing may suffer from the granularity of the blocks.

---

[2]An octree is a tree data structure in which each internal node has exactly eight children. Octrees are most often used to partition a three-dimensional space by recursively subdividing it into eight octants.

c. Parallel particle dynamics in irregular domains

Multiple techniques for parallel particle dynamics have been suggested over the years. In particular, PD has now reached a good degree of efficiency when dealing with regular geometries. However, in the presence of highly irregular domains, such as those found in biofluidic devices or in physiological conduits, several critical issues arise related to the calculation of forces and migration of particles among subdomains. For instance, irregular subdomains imply irregular contact surfaces and, in principle, irregular communication patterns. The geometrical tests for particle ownership and the exchange of particles among domains requires strategic decisions that affect the efficiency of stand-alone PD as much as the LBPD multiphysics applications.

The first general solution to those problem was presented by Bisson, Bernaschi, and Melchionna (2011). The proposed method relies on two basic notions: proximity and membership tests. Those tests are used to discriminate particles according to their position relative to the geometry of the domains. Proximity tests are used to select the particles that have out-of-domain interactions and are used to perform interdomain forces computation. The membership tests regard the assignment of particles to domains and exploit a tracking method to associate particles position to the domains morphology. Moreover, the critical regions around the contact surfaces of the subdomains are approximated so that it is computationally simple to find a superset of the particles located inside those regions and to apply the tests only to those particles. This is possible by covering each subdomain with identical box-shaped cells.

There are several other issues that deserve attention in the design and implementation of large-scale LBPD based simulations. However, for brevity, here we just mention them: (i) whether it is better to use only accelerators to run the simulation leaving the CPU as a sort of communication and I/O coprocessor of the accelerators or if it makes sense to develop hybrid codes running the LBPD code on both the CPU and the accelerators; (ii) whether it is actually possible to develop portable LB high-performance codes by using directive-based software; and finally, (iii) how to implement some form of fault tolerance within the LBPD, so as to secure prompt and error-free recovery from hardware or software failures in multimillion and possibly billion-core computing environments.

REFERENCES

Abraham, F. F., J. Q. Broughton, N. Bernstein, and E. Kaxiras, 1998, "Spanning the continuum to quantum length scales in a dynamic simulation of brittle fracture," Europhys. Lett. **44**, 783.

Adhikari, R., K. Stratford, M. E. Cates, and A. J. Wagner, 2005, "Fluctuating lattice boltzmann," Europhys. Lett. **71**, 473.

Ahlrichs, P., and B. Dünweg, 1998, "Lattice-Boltzmann simulation of polymer-solvent systems," Int. J. Mod. Phys. C **09**, 1429–1438.

Alfahani, F., M. Antonelli, and J. Kreft Pearce, 2015, "Separation of DNA by length in rotational flow: Lattice-Boltzmann-based simulations," Biomicrofluidics **9**, 044107.

Alowayyed, S., D. Groen, P. V. Coveney, and A. G. Hoekstra, 2017, "Multiscale computing in the exascale era," J. Comput. Sci. **22**, 15–25.

Anderson, C., 2008, "The end of theory: The data deluge makes the scientific method obsolete. Wired Magazine 16.07," https://www.wired.com/2008/06/pb-theory/.

Ansumali, S., and I. V. Karlin, 2002, "Kinetic boundary conditions in the lattice Boltzmann method," Phys. Rev. E **66**, 026311.

Ansumali, S., I. V. Karlin, S. Arcidiacono, A. Abbas, and N. I. Prasianakis, 2007, "Hydrodynamics beyond Navier-Stokes: Exact solution to the lattice Boltzmann hierarchy," Phys. Rev. Lett. **98**, 124502.

Artoli, A. M., A. G. Hoekstra, and P. M. A. Sloot, 2003, "Simulation of a systolic cycle in a realistic artery with the Lattice Boltzmann BGK method," Int. J. Mod. Phys. B **17**, 95–98.

Auer, S., et al., 2008, "A generic mechanism of emergence of amyloid protofilaments from disordered oligomeric aggregates," PLoS Comput. Biol. **4**, e1000222.

Axner, L., J. Bernsdorf, T. Zeiser, P. Lammers, J. Linxweiler, and A. G. Hoekstra, 2008, "Performance evaluation of a parallel sparse lattice Boltzmann solver," J. Comput. Phys. **227**, 4895–4911.

Ayodele, S. G., F. Varnik, and D. Raabe, 2011, "Lattice Boltzmann study of pattern formation in reaction-diffusion systems," Phys. Rev. E **83**, 016702.

Bekard, I. B., P. Asimakis, J. Bertolini, and D. E. Dunstan, 2011, "The effects of shear flow on protein structure and function," Biopolymers **95**, 733–745.

Benzi, R., S. Succi, and M. Vergassola, 1992, "The lattice Boltzmann equation: Theory and applications," Phys. Rep. **222**, 145–197.

Bernaschi, M., M. Bisson, T. Endo, S. Matsuoka, and M. Fatica, 2011, "Petaflop biofluidics simulations on a two million-core system," in *2011 International Conference for High Performance Computing, Networking, Storage and Analysis (SC)* (IEEE, New York), pp. 1–12.

Bernaschi, M., M. Bisson, M. Fatica, and S. Melchionna, 2013a, "20 petaflops simulation of proteins suspensions in crowding conditions," in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis* (ACM, New York), p. 2.

Bernaschi, M., M. Bisson, M. Fatica, S. Melchionna, and S. Succi, 2013b, "Petaflop hydrokinetic simulations of complex flows on massive GPU clusters," Comput. Phys. Commun. **184**, 329–341.

Bernaschi, Massimo, Massimiliano Fatica, Simone Melchionna, Sauro Succi, and Efthimios Kaxiras, 2010, "A flexible high-performance Lattice Boltzmann GPU code for the simulations of fluid flows in complex geometries," *Concurrency and Computation: Practice and Experience* (Wiley Online Library), Vol. 22, pp. 1–14.

Bernaschi, M., S. Melchionna, S. Succi, M. Fyta, and E. Kaxiras, 2008, "Quantized current blockade and hydrodynamic correlations in biopolymer translocation through nanopores: Evidence from multiscale simulations," Nano Lett. **8**, 1115–1119.

Bernaschi, M., S. Melchionna, S. Succi, M. Fyta, E. Kaxiras, and J, K. Sircar, 2009, "MUPHY: A parallel MUlti PHYsics/scale code for high performance bio-fluidic simulations," Comput. Phys. Commun. **180**, 1495–1502.

Bhatnagar, P. L., E. P. Gross, and M. Krook, 1954, "A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems," Phys. Rev. **94**, 511.

Bird, G. A., 1994, "Molecular gas dynamics and the direct simulation Monte Carlo of gas flows," *Oxford Engineering Science Series* (Clarendon Press, Oxford), Vol. 508, p. 128.

Bisson, M., M. Bernaschi, and S. Melchionna, 2011, "Parallel Molecular Dynamics with Irregular Domain Decomposition," Commun. Comput. Phys. **10**, 1071–1088.

Bobylev, A. V., 1982, "The Chapman-Enskog and Grad methods for solving the Boltzmann equation," Akad. Nauk SSSR Dokl. **262**, 71–75.

Boltzmann, Ludwig, 2012, *Lectures on gas theory* (Courier Corporation, North Chelmsford, MA).

Boon, J. P., D. Dab, K. Raymond, and L. Anna, 1996, "Lattice gas automata for reactive systems," Phys. Rep. **273**, 55–147.

Boon, J. P., and S. Yip, 1991, *Molecular hydrodynamics* (Courier Corporation, North Chelmsford, MA).

Boyd, J., J. M. Buick, and S. Green, 2007, "Analysis of the Casson and Carreau-Yasuda non-Newtonian blood models in steady and oscillatory flows using the lattice Boltzmann method," Phys. Fluids **19**, 093103.

Bryngelson, J. D., J. N. Onuchic, N. D. Socci, and P. G. Wolynes, 1995, "Funnels, pathways, and the energy landscape of protein folding: A synthesis," *Proteins: Structure, Function, and Bioinformatics* (Wiley Online Library), Vol. 21, pp. 167–195.

Buckner, R. L., *et al.*, 2005, "Molecular, structural, and functional characterization of Alzheimer's disease: Evidence for a relationship between default activity, amyloid, and memory," J. Neurosci. **25**, 7709–7717.

Buick, J. M., J. A. Cosgrove, S. J. Tonge, M. W. Collins, A. J. Mulholland, and B. A. Steves, 2003, "The Lattice Boltzmann equation for modelling arterial flows: Review and application," Biomed. Pharmacother. **56**, 345–346.

Caiazzo, A., *et al.*, 2009, "Towards a complex automata multiscale model of in-stent restenosis," in *International Conference on Computational Science* (Springer, New York), pp. 705–714.

Capuani, F., I. Pagonabarraga, and D. Frenkel, 2004, "Discrete solution of the electrokinetic equations," J. Chem. Phys. **121**, 973–986.

Cercignani, C., and A. S. Berman, 1976, "Theory and application of the Boltzmann equation," J. Appl. Mech. **43**, 521.

Chebaro, Y., S. Pasquali, and P. Derreumaux, 2012, "The Coarse-Grained OPEP Force Field for Non-Amyloid and Amyloid Proteins," J. Phys. Chem. B **116**, 8741–8752.

Chen, H., O. Filippova, J. Hoch, K. Molvig, R. Shock, C. Teixeira, and R. Zhang, 2006, "Grid refinement in lattice Boltzmann methods based on volumetric formulation," Physica A (Amsterdam) **362**, 158–167.

Chen, H., S. Kandasamy, S. Orszag, R. Shock, S. Succi, and V. Yakhot, 2003, "Extended Boltzmann kinetic equation for turbulent flows," Science **301**, 633–636.

Chen, S., D. Martinez, and R. Mei, 1996, "On boundary conditions in lattice Boltzmann methods," Phys. Fluids **8**, 2527–2536.

Chen, Y.-L., H. Ma, M. D. Graham, and J. J. De Pablo, 2007, "Modeling DNA in confinement: A comparison between the Brownian dynamics and lattice Boltzmann method," Macromolecules **40**, 5978–5984.

Chinappi, M., C. M. Casciola, F. Cecconi, U. M. B. Marconi, and S. Melchionna, 2014, "Modulation of current through a nanopore induced by a charged globule: Implications for DNA-docking," Europhys. Lett. **108**, 46002.

Chiricotto, M., S. Melchionna, P. Derreumaux, and F. Sterpone, 2016, "Hydrodynamic effects on β-amyloid (16–22) peptide aggregation," J. Chem. Phys. **145**, 035102.

Chiricotto, M., F. Sterpone, P. Derreumaux, and S. Melchionna, 2016, "Multiscale simulation of molecular processes in cellular environments," Phil. Trans. R. Soc. A **374**, 20160225.

Chiricotto, M., T. T. Tran, P. H. Nguyen, S. Melchionna, F. Sterpone, and P. Derreumaux, 2017, "Coarse-grained and All-atom Simulations towards the Early and Late Steps of Amyloid Fibril Formation," Isr. J. Chem. **57**, 564–573.

Chopard, B., and M. Droz, 1998, *Cellular automata* (Springer, New York).

Chopard, B., R. Ouared, and D. A. Rüfenacht, 2006, "A lattice Boltzmann simulation of clotting in stented aneursysms and comparison with velocity or shear rate reductions," Math. Comput. Simul. **72**, 108–112.

Cuda C Programming Guide, 2019, https://docs.nvidia.com/cuda/cuda-c-programming-guide.

Clausen, J. R., D. A. Reasor, and C. K. Aidun, 2010, "Parallel performance of a lattice-Boltzmann/finite element cellular blood flow solver on the IBM Blue Gene/P architecture," Comput. Phys. Commun. **181**, 1013–1020.

Courant, R., K. Friedrichs, and H. Lewy, 1928, "Über die partiellen Differenzengleichungen der mathematischen Physik," Math. Ann. **100**, 32–74.

Coveney, P. V., J. P. Boon, and S. Succi, 2016, *Bridging the gaps at the physics–chemistry–biology interface* (The Royal Society, London).

Coveney, P. V., E. R. Dougherty, and R. R. Highfield, 2016, "Big data need big theory too," Phil. Trans. R. Soc. A **374**, 20160153.

Cruz-León, S., A. Vázquez-Mayagoitia, S. Melchionna, N. Schwierz, and M. Fyta, 2018, "A Coarse-Grained Double-Stranded RNA Model from Quantum-Mechanical Calculations," J. Phys. Chem. B **122**, 7915–7928.

Datar, A. V., M. Fyta, U. M. B. Marconi, and S. Melchionna, 2017, "Electrokinetic Lattice Boltzmann solver coupled to Molecular Dynamics: Application to polymer translocation," Langmuir **33**, 11635–11645.

De Rosis, A., 2014, "Analysis of blood flow in deformable vessels via a lattice Boltzmann approach," Int. J. Mod. Phys. C **25**, 1350107.

Descovich, X., G. Pontrelli, S. Melchionna, S. Succi, and S. Wassertheurer, 2013, "Modeling fluid flows in distensible tubes for applications in hemodynamics," Int. J. Mod. Phys. C **24**, 1350030.

d'Humières, D., 1992, "Generalized Lattice Boltzmann Equations, Rarefied Gas Dynamics: Theory and Simulations," Prog. Astronaut. Aeronaut. **159**, 450–458.

Di Ilio, G., D. Chiappini, S. Ubertini, G. Bella, and S. Succi, 2017, "Hybrid lattice Boltzmann method on overlapping grids," Phys. Rev. E **95**, 013309.

Dimarco, G., R. Loubére, J. Narski, and T. Rey, 2018, "An efficient numerical method for solving the Boltzmann equation in multi-dimensions," J. Comput. Phys. **353**, 46–81.

Doyle, D. A., J. M. Cabral, R. A. Pfuetzner, A. Kuo, J. M. Gulbis, S. L. Cohen, B. T. Chait, and R. MacKinnon, 1998, "The structure of the potassium channel: Molecular basis of K+ conduction and selectivity," Science **280**, 69–77.

Dunstan, D. E., P. Hamilton-Brown, P. Asimakis, W. Ducker, and J. Bertolini, 2009, "Shear flow promotes amyloid-β fibrilization," Protein Engineering, Design & Selection **22**, 741–746.

Dünweg, B., and A. J. Ladd, 2009, "Lattice Boltzmann simulations of soft matter systems," in *Advanced Computer Simulation Approaches for Soft Matter Sciences III* (Springer, New York), pp. 89–166.

Dünweg, B., U. D. Schiller, and A. J. Ladd, 2007, "Statistical mechanics of the fluctuating lattice Boltzmann equation," Phys. Rev. E **76**, 036704.

Dupin, M. M., I. Halliday, and C. M. Care, 2003, "Multi-component lattice Boltzmann equation for mesoscale blood flow," J. Phys. A **36**, 8517.

Dupin, M. M., I. Halliday, and C. M. Care, 2006, "A multi-component lattice Boltzmann scheme: Towards the mesoscale simulation of blood flow," Medical engineering & physics **28**, 13–18.

Dupin, M. M., I. Halliday, C. M. Care, and L. L. Munn, 2008, "Lattice Boltzmann modelling of blood cell dynamics," Int. J. Comput. Fluid Dyn. **22**, 481–492.

Dupuis, A., and B. Chopard, 1999, "Lattice gas: An efficient and reusable parallel library based on a graph partitioning technique," in *International Conference on High-Performance Computing and Networking* (Springer, New York), pp. 319–328.

Eitel, G., R. K. Freitas, A. Lintermann, M. Meinke, and W. Schröder, 2010, "Numerical simulation of nasal cavity flow based on a lattice-boltzmann method," in *New Results in Numerical and Experimental Fluid Mechanics VII* (Springer, New York), pp. 513–520.

Ellis, R. J., 2001, "Macromolecular crowding: Obvious but under-appreciated," Trends Biochem. Sci. **26**, 597–604.

Falcucci, G., *et al.*, 2016, "Mapping reactive flow patterns in monolithic nanoporous catalysts," Microfluid. Nanofluid. **20**, 105.

Fang, H., Z. Wang, Z. Lin, and M. Liu, 2002, "Lattice Boltzmann method for simulating the viscous flow in large distensible blood vessels," Phys. Rev. E **65**, 051925.

Fedosov, D. A., B. Caswell, and G. E. Karniadakis, 2010, "A multiscale red blood cell model with accurate mechanics, rheology, and dynamics," Biophys. J. **98**, 2215–2225.

Feichtinger, C., S. Donath, H. Köstler, J. Götz, and U. Rüde, 2011, "WaLBerla: HPC software design for computational engineering simulations," J. Comput. Sci. **2**, 105–112.

Feig, M., I. Yu, P.-h. Wang, G. Nawrocki, and Y. Sugita, 2017, "Crowding in Cellular Environments at an Atomistic Level from Computer Simulations," J. Phys. Chem. B **121**, 8009.

Fenner, J. W., *et al.*, 2008, "The EuroPhysiome, STEP and a roadmap for the virtual physiological human," Phil. Trans. R. Soc. A **366**, 2979–2999.

Fersht, A., 2017, *Structure and mechanism in protein science: A Guide to enzyme catalysis and protein folding* (World Scientific, Singapore), Vol. 9.

Frauenfelder, H., S. G. Sligar, and P. G. Wolynes, 1991, "The energy landscapes and motions of proteins," Science **254**, 1598–1603.

Freitas, R. K., and W. Schröder, 2008, "Numerical investigation of the three-dimensional flow in a human lung model," J. Biomech. **41**, 2446–2457.

Fyta, M., S. Melchionna, E. Kaxiras, and S. Succi, 2008, "Multiscale simulation of nanobiological flows," Comput. Sci. Eng. **10**, 10.

Fyta, M., S. Melchionna, and S. Succi, 2011, "Translocation of biomolecules through solid-state nanopores: Theory meets experiments," J. Polym. Sci., Part B: Polym. Phys. **49**, 985–1011.

Fyta, M., S. Melchionna, S. Succi, and E. Kaxiras, 2008, "Hydrodynamic correlations in the translocation of a biopolymer through a nanopore: Theory and multiscale simulations," Phys. Rev. E **78**, 036704.

Fyta, M. G., S. Melchionna, E. Kaxiras, and S. Succi, 2006, "Multiscale coupling of molecular dynamics and hydrodynamics: Application to DNA translocation through a nanopore," Multiscale Modeling & Simulation **5**, 1156–1173.

Gan, Y., A. Xu, G. Zhang, and S. Succi, 2015, "Discrete Boltzmann modeling of multiphase flows: Hydrodynamic and thermodynamic non-equilibrium effects," Soft Matter **11**, 5336–5345.

Goldberg, D. E., and J. H. Holland, 1988, "Genetic algorithms and machine learning," Mach. Learn. **3**, 95–99.

Grad, H., 1949, "On the kinetic theory of rarefied gases," Commun. Pure Appl. Math. **2**, 331–407.

Groen, D., J. Hetherington, H. B. Carver, R. W. Nash, M. O. Bernabeu, and P. V. Coveney, 2013, "Analysing and modelling the performance of the HemeLB lattice-Boltzmann simulation environment," J. Comput. Sci. **4**, 412–422.

Gunstensen, A. K., D. H. Rothman, S. Zaleski, and G. Zanetti, 1991, "Lattice Boltzmann model of immiscible fluids," Phys. Rev. A **43**, 4320.

Guvench, O., and A. D. MacKerell, 2008, "Comparison of protein force fields for molecular dynamics simulations," Molecular modeling of proteins (Springer, New York), pp. 63–88.

Hammack, A., Y.-L. Chen, and J. K. Pearce, 2011, "Role of dissolved salts in thermophoresis of DNA: Lattice-Boltzmann-based simulations," Phys. Rev. E **83**, 031915.

Hansen, J.-P., and I. R. McDonald, 1990, *Theory of simple liquids* (Elsevier, New York).

Harrison, S. E., J. Bernsdorf, D. R. Hose, and P. V. Lawford, 2008, "A lattice Boltzmann framework for simulation of thrombogenesis," Prog. Comput. Fluid Dyn. **8**, 121–128.

Harrison, S. E., S. M. Smith, J. Bernsdorf, D. R. Hose, and P. V. Lawford, 2007, "Application and validation of the lattice Boltzmann method for modelling flow-related clotting," J. Biomech. **40**, 3023–3028.

Hénon, M., 1987, "Viscosity of a lattice gas," Complex Syst. **1**, 762–790.

Heuveline, V., and J. Latt, 2007, "The OpenLB project: An open source and object oriented implementation of lattice Boltzmann methods," Int. J. Mod. Phys. C **18**, 627–634.

Hickey, O. A., C. Holm, and J. Smiatek, 2014, "Lattice-Boltzmann simulations of the electrophoretic stretching of polyelectrolytes: The importance of hydrodynamic interactions," J. Chem. Phys. **140**, 164904.

Higuera, F. J., S. Succi, and R. Benzi, 1989, "Lattice gas dynamics with enhanced collisions," Europhys. Lett. **9**, 345.

Hille, B., *et al.*, 2001, *Ion channels of excitable membranes* (Sinauer Sunderland, MA), Vol. 507.

Hirabayashi, M., M. Ohta, D. A. Rüfenacht, and B. Chopard, 2004, "A lattice Boltzmann study of blood flow in stented aneurism," *Future Generation Computer Systems* (Elsevier, New York), Vol. 20, pp. 925–934.

Hoekstra, A. G., J. van't Hoff, A. M. M. Artoli, and P. M. Sloot, 2003, "Lattice BGK simulations of unsteady flow in a 2D elastic tube," in *International Conference on Computational Science* (Springer, New York), pp. 997–1006.

Horbach, J., and S. Succi, 2006, "Lattice Boltzmann versus molecular dynamics simulation of nanoscale hydrodynamic flows," Phys. Rev. Lett. **96**, 224503.

Hosni, Hykel, and Angelo Vulpiani, 2018, "Forecasting in light of big data," *Philosophy & Technology* (Springer, New York), Vol. 31, pp. 557–569.

Hsu, C. W., M. Fyta, G. Lakatos, S. Melchionna, and E. Kaxiras, 2012, "*Ab initio* determination of coarse-grained interactions in double-stranded DNA," J. Chem. Phys. **137**, 105102.

Hua-Bing, L., J. Li, and Q. Bing, 2008, "Deformation of two-dimensional nonuniform-membrane red blood cells simulated by a lattice Boltzmann model," Chin. Phys. Lett. **25**, 4042.

Janoschek, F., F. Toschi, and J. Harting, 2010, "Simplified particulate model for coarse-grained hemodynamics simulations," Phys. Rev. E **82**, 056710.

Junk, M., A. Klar, and L.-S. Luo, 2005, "Asymptotic analysis of the lattice Boltzmann equation," J. Comput. Phys. **210**, 676–704.

Kamerlin, S. C., S. Vicatos, A. Dryga, and A. Warshel, 2011, "Coarse-Grained (Multiscale) Simulations in Studies of Biophysical and Chemical Systems," Annu. Rev. Phys. Chem. 62, 41–64.

Karlin, I. V., A. Ferrante, and H. C. Öttinger, 1999, "Perfect entropy functions of the lattice Boltzmann method," Europhys. Lett. 47, 182.

Karlin, I. V., A. N. Gorban, S. Succi, and V. Boffi, 1998, "Maximum entropy principle for lattice kinetic equations," Phys. Rev. Lett. 81, 6.

Karpatne, A., W. Watkins, J. Read, and V. Kumar, 2017, "Physics-guided Neural Networks (PGNN): An Application in Lake Temperature Modeling," arXiv:1710.11431.

Keller, S. R., and R. Skalak, 1982, "Motion of a tank-treading ellipsoidal particle in a shear flow," J. Fluid Mech. 120, 27–47.

Ketsdever, A., and H. Struchtrup, 2016, "30th International Symposium on Rarefied Gas Dynamics: RGD 30 1786," (IOP Publishing, Bristol, UK).

Knowles, T. P. J., M. Vendruscolo, and C. M. Dobson, 2015, "The physical basis of protein misfolding disorders," Phys. Today 68, No. 3, 36.

Krause, M. J., 2010, "Fluid flow simulation and optimisation with lattice Boltzmann methods on high performance computers: Application to the human respiratory system," Ph.D. thesis (Karlsruhe Institute of Technology).

Krüger, T., M. Gross, D. Raabe, and F. Varnik, 2013, "Crossover from tumbling to tank-treading-like motion in dense simulated suspensions of red blood cells," Soft Matter 9, 9008–9015.

Krüger, T., D. Holmes, and P. V. Coveney, 2014, "Deformability-based red blood cell separation in deterministic lateral displacement devices—A simulation study," Biomicrofluidics 8, 054114.

Krüger, T., H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, and E. M. Viggen, 2017, The Lattice Boltzmann Method: Principles and Practice (Springer, New York).

Ladd, A. J. C., 1993, "Short-time motion of colloidal particles: Numerical simulation via a fluctuating lattice-Boltzmann equation," Phys. Rev. Lett. 70, 1339.

Ladd, A. J. C., 1994a, "Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation," J. Fluid Mech. 271, 285–309.

Ladd, A. J. C., 1994b, "Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results," J. Fluid Mech. 271, 311–339.

Lagrava, D., O. Malaspinas, J. Latt, and B. Chopard, 2012, "Advances in multi-domain lattice Boltzmann grid refinement," J. Comput. Phys. 231, 4808–4822.

Latt, J., and B. Chopard, 2006, "Lattice Boltzmann method with regularized pre-collision distribution functions," Math. Comput. Simul. 72, 165–168.

Leclaire, S., A. Parmigiani, O. Malaspinas, B. Chopard, and J. Latt, 2017, "Generalized three-dimensional lattice Boltzmann color-gradient method for immiscible two-phase pore-scale imbibition and drainage in porous media," Phys. Rev. E 95, 033306.

Ledesma-Aguilar, R., T. Sakaue, and J. M. Yeomans, 2012, "Easier sieving through narrower pores: Fluctuations and barrier crossing in flow-driven polymer translocation," Soft Matter 8, 4306–4309.

Lee, T., and C.-L. Lin, 2005, "A stable discretization of the lattice Boltzmann equation for simulation of incompressible two-phase flows at high density ratio," J. Comput. Phys. 206, 16–47.

Levitt, M., and A. Warshel, 1975, "Computer simulation of protein folding," Nature (London) 253, 694.

Li, Z., and C. Kleinstreuer, 2011, "Airflow analysis in the alveolar region using the lattice-Boltzmann method," Med. Biol. Eng. Comput. 49, 441–451.

Lindorff-Larsen, K., P. Maragakis, S. Piana, M. P. Eastwood, R. O. Dror, and D. E. Shaw, 2012, "Systematic validation of protein force fields against experimental data," PLoS One 7, e32131.

Lintermann, A., M. Meinke, and W. Schröder, 2011, "Investigations of human nasal cavity flows based on a Lattice-Boltzmann method," in High Performance Computing on Vector Systems 2011 (Springer, New York), pp. 143–158.

Liu, Y., L. Moevius, X. Xu, T. Qian, J. M. Yeomans, and Z. Wang, 2014, "Pancake bouncing on superhydrophobic surfaces," Nat. Phys. 10, 515–519.

Luo, L.-S., 2004, "Comment on Discrete Boltzmann equation for microfluidics," Phys. Rev. Lett. 92, 139401.

MacMeccan, R. M., J. R. Clausen, G. P. Neitzel, and C. K. Aidun, 2009, "Simulating deformable particle suspensions using a coupled lattice-Boltzmann and finite-element method," J. Fluid Mech. 618, 13–39.

Marconi, U. M. B., and S. Melchionna, 2009, "Kinetic theory of correlated fluids: From dynamic density functional to Lattice Boltzmann methods," J. Chem. Phys. 131, 014105.

Marconi, U. M. B., and S. Melchionna, 2011a, "Dynamics of fluid mixtures in nanospaces," J. Chem. Phys. 134, 064118.

Marconi, U. M. B., and S. Melchionna, 2011b, "Multicomponent diffusion in nanosystems," J. Chem. Phys. 135, 044104.

Marconi, U. M. B., and S. Melchionna, 2012, "Charge transport in nanochannels: A molecular theory," Langmuir 28, 13727–13740.

Marconi, U. M. B., S. Melchionna, and I. Pagonabarraga, 2013, "Effective electrodiffusion equation for non-uniform nanochannels," J. Chem. Phys. 138, 244107.

Masliyah, J. H., and S. Bhattacharjee, 2006, Electrokinetic and colloid transport phenomena (John Wiley & Sons, New York).

Matyka, M., Z. Koza, and Ł. Mirosław, 2013, "Wall orientation and shear stress in the lattice Boltzmann model," Comput. Fluids 73, 115–123.

Maxwell, J. C., 1878, "On Stresses in Rarefied Gases Arising from Inequalities of Temperature.," Proc. R. Soc. London 27, 304–308.

Mazzeo, M. D., and P. V. Coveney, 2008, "HemeLB: A high performance parallel lattice-Boltzmann code for large scale fluid flow in complex geometries," Comput. Phys. Commun. 178, 894–914.

McWhirter, J. L., H. Noguchi, and G. Gompper, 2009, "Flow-induced clustering and alignment of vesicles and red blood cells in microcapillaries," Proc. Natl. Acad. Sci. U.S.A. 106, 6039–6043.

Melchionna, S., 2011, "A Model for Red Blood Cells in Simulations of Large-scale Blood Flows," Macromol. Theory Simul. 20, 548–561.

Melchionna, S., and U. M. B. Marconi, 2008, "Lattice Boltzmann method for inhomogeneous fluids," Europhys. Lett. 81, 34001.

Melchionna, S., M. Bernaschi, M. Fyta, E. Kaxiras, and S. Succi, 2009, "Quantized biopolymer translocation through nanopores: Departure from simple scaling," Phys. Rev. E 79, 030901.

Melchionna, S., M. Bernaschi, S. Succi, E. Kaxiras, F. J. Rybicki, D. Mitsouras, A. U. Coskun, and C. L. Feldman, 2010, "Hydrokinetic approach to large-scale cardiovascular blood flow," Comput. Phys. Commun. 181, 462–472.

Melchionna, S., M. G. Fyta, E. Kaxiras, and S. Succi, 2007, "Exploring DNA translocation through a nanopore via a multiscale Lattice-Boltzmann molecular-dynamics methodology," Int. J. Mod. Phys. C 18, 685–692.

Melchionna, S., and U. M. B. Marconi, 2011, "Electro-osmotic flows under nanoconfinement: A self-consistent approach," Europhys. Lett. 95, 44002.

Melchionna, S., S. Succi, and J.-P. Hansen, 2006, "Simulation of single-file ion transport with the lattice Fokker-Planck equation," Phys. Rev. E **73**, 017701.

Mendoza, M., B. M. Boghosian, H. J. Herrmann, and S. Succi, 2010, "Derivation of the lattice Boltzmann model for relativistic hydrodynamics," Phys. Rev. D **82**, 105008.

Mendoza, M., S. Succi, and H. J. Herrmann, 2014, "Kinetic formulation of the Kohn-Sham equations for *ab initio* electronic structure calculations," Phys. Rev. Lett. **113**, 096402.

Meng, J., and Y. Zhang, 2011, "Gauss-Hermite quadratures and accuracy of lattice Boltzmann models for nonequilibrium gas flows," Phys. Rev. E **83**, 036704.

Miki, T., X. Wang, T. Aoki, Y. Imai, T. Ishikawa, K. Takase, and T. Yamaguchi, 2012, "Patient-specific modelling of pulmonary airflow using GPU cluster for the application in medical practice," Comput. Methods Biomech. Biomed. Eng. **15**, 771–778.

Miocchi, P., P. Derreumaux, F. Sterpone, and S. Melchionna, 2019, "Mesoscale biosimulations within a unified framework: From proteins to plasmids," Mol. Simul., 1–12.

Montessori, A., G. Falcucci, P. Prestininzi, M. La Rocca, and S. Succi, 2014, "Regularized lattice Bhatnagar-Gross-Krook model for two-and three-dimensional cavity flow simulations," Phys. Rev. E **89**, 053317.

Montessori, A., P. Prestininzi, M. La Rocca, and S. Succi, 2017, "Entropic lattice pseudo-potentials for multiphase flow simulations at high Weber and Reynolds numbers," Phys. Fluids **29**, 092103.

Montessori, A., and G. Falcucci, 2018, *Lattice Boltzmann Modeling of Complex Flows for Engineering Applications* (Morgan & Claypool Publishers, San Rafael, CA).

Montessori, A., M. Lauricella, M. La Rocca, S. Succi, E. Stolovicki, R. Ziblat, and D. Weitz, 2018, "Regularized lattice Boltzmann multicomponent models for low capillary and Reynolds microfluidics flows," Comput. Fluids **167**, 33–39.

Montessori, A., M. Lauricella, and S. Succi, 2018, "Mesoscale modelling of soft flowing crystals," arXiv:1807.05415.

Montessori, A., M. Lauricella, S. Succi, E. Stolovicki, and D. Weitz, 2018, "Elucidating the mechanism of step emulsification," Phys. Rev. Fluids **3**, 072202.

Moroni, D., B. Rotenberg, J.-P. Hansen, S. Succi, and S. Melchionna, 2006, "Solving the Fokker-Planck kinetic equation on a lattice," Phys. Rev. E **73**, 066707.

Munn, L. L., and M. M. Dupin, 2008, "Blood cell interactions and segregation in flow," Ann. Biomed. Eng. **36**, 534–544.

Nasica-Labouze, Jessica, *et al.*, 2015, "Amyloid β-protein and Alzheimer's Disease: When Computer Simulations Complement Experimental Studies," Chem. Rev. **115**, 3518.

Nie, X., X. Shan, and H. Chen, 2008, "Thermal lattice Boltzmann model for gases with internal degrees of freedom," Phys. Rev. E **77**, 035701.

Noble, D., 2008, *The music of life: Biology beyond genes* (Oxford University Press, New York).

Noble, D., 2016, *Dance to the Tune of Life: Biological Relativity* (Cambridge University Press, Cambridge, England).

Noguchi, H., and G. Gompper, 2005, "Shape transitions of fluid vesicles and red blood cells in capillary flows," Proc. Natl. Acad. Sci. U.S.A. **102**, 14159–14164.

Noid, W. G., J.-W. Chu, G. S. Ayton, V. Krishna, S. Izvekov, G. A. Voth, A. Das, and H. C. Andersen, 2008, "The multiscale coarse-graining method. I. A rigorous bridge between atomistic and coarse-grained models," J. Chem. Phys. **128**, 244114.

Omori, T., Y. Imai, K. Kikuchi, T. Ishikawa, and T. Yamaguchi, 2015, "Hemodynamics in the microcirculation and in microfluidics," Ann. Biomed. Eng. **43**, 238–257.

Ouared, R., and B. Chopard, 2005, "Lattice Boltzmann simulations of blood flow: Non-Newtonian rheology and clotting processes," J. Stat. Phys. **121**, 209–221.

Ouared, R., B. Chopard, B. Stahl, D. A. Rüfenacht, H. Yilmaz, and G. Courbebaisse, 2008, "Thrombosis modeling in intracranial aneurysms: A lattice Boltzmann numerical algorithm," Comput. Phys. Commun. **179**, 128–131.

Papoian, G. A., J. Ulander, M. P. Eastwood, Z. Luthey-Schulten, and P. G. Wolynes, 2004, "Water in protein structure prediction," Proc. Natl. Acad. Sci. U.S.A. **101**, 3352–3357.

Patronis, A., R. A. Richardson, S. Schmieschek, B. J. Wylie, R. W. Nash, and P. V. Coveney, 2018, "Modelling Patient-Specific Magnetic Drug Targeting within the Intracranial Vasculature," Front. Physiol. **9**, 331.

Pelliccioni, O., M. Cerrolaza, and M. Herrera, 2007, "Lattice Boltzmann dynamic simulation of a mechanical heart valve device," Math. Comput. Simul. **75**, 1–14.

Peskin, C. S., 2002, "The immersed boundary method," Acta Numer. **11**, 479–517.

Ponce Dawson, S., S. Chen, and G. D. Doolen, 1993, "Lattice Boltzmann computations for reaction-diffusion equations," J. Chem. Phys. **98**, 1514–1523.

Ponder, J. W., and D. A. Case, 2003, "Force fields for protein simulations," Adv. Protein Chem. **66**, 27–85.

Pontrelli, G., I. Halliday, S. Melchionna, T. J. Spencer, and S. Succi, 2012, "The Lattice Boltzmann Method and Multiscale Hemodynamics: Recent Advances and Perspectives," *IFAC Proceedings Volumes* (Elsevier, New York), Vol. 45, pp. 30–39.

Pontrelli, G., I. Halliday, S. Melchionna, T. J. Spencer, and S. Succi, 2014, "Lattice Boltzmann method as a computational framework for multiscale haemodynamics," Mathematical and Computer Modelling of Dynamical Systems **20**, 470–490.

Pontrelli, G., I. Halliday, T. J. Spencer, C. S. König, and M. W. Collins, 2015, "Modelling the glycocalyx–endothelium–erythrocyte interaction in the microcirculation: A computational study," Comput. Methods Biomech. Biomed. Eng. **18**, 351–361.

Praprotnik, M., L. D. Site, and K. Kremer, 2008, "Multiscale simulation of soft matter: From scale bridging to adaptive resolution," Annu. Rev. Phys. Chem. **59**, 545–571.

Prigogine, I., 2017, *Non-equilibrium statistical mechanics* (Courier Dover Publications, New York).

Purcell, E. M., 1977, "Life at low Reynolds number," Am. J. Phys. **45**, 3–11.

Qian, Y. H., D. d'Humières, and P. Lallemand, 1992, "Lattice BGK models for Navier-Stokes equation," Europhys. Lett. **17**, 479.

Raissi, M., and G. E. Karniadakis, 2018, "Hidden physics models: Machine learning of nonlinear partial differential equations," J. Comput. Phys. **357**, 125–141.

Reasor, D. A., J. R. Clausen, and C. K. Aidun, 2012, "Coupling the lattice-Boltzmann and spectrin-link methods for the direct numerical simulation of cellular blood flow," Int. J. Numer. Methods Fluids **68**, 767–781.

Reboux, S., F. Capuani, N. González-Segredo, and D. Frenkel, 2006, "Lattice-Boltzmann simulations of ionic current modulation by DNA translocation," J. Chem. Theory Comput. **2**, 495–503.

Rivet, J.-P., and J. P. Boon, 2005, *Lattice gas hydrodynamics* (Cambridge University Press, Cambridge, England), Vol. 11.

Rossinelli, D., *et al.*, 2015, "The in-silico lab-on-a-chip: Petascale and high-throughput simulations of microfluidics at cell resolution," in *Proceedings of the International Conference for High*

Performance Computing, Networking, Storage and Analysis (ACM, New York), p. 2.

Rybicki, Frank J., *et al.*, 2009, "Prediction of coronary artery plaque progression and potential rupture from 320-detector row prospectively ECG-gated single heart beat CT angiography: Lattice Boltzmann evaluation of endothelial shear stress," Int. J. Cardiovascular Imaging **25**, 289–299.

Sbragaglia, M., H. Chen, X. Shan, and S. Succi, 2009, "Continuum free-energy formulation for a class of lattice Boltzmann multiphase models," Europhys. Lett. **86**, 24005.

Schulz, M., M. Krafczyk, J. Tölke, and E. Rank, 2002, "Parallelization strategies and efficiency of CFD computations in complex geometries using Lattice Boltzmann methods on high-performance computers," *High performance scientific and engineering computing* (Springer, New York), Vol. 21, pp. 115–122.

Sega, M., M. Sbragaglia, S. S. Kantorovich, and A. O. Ivanov, 2013, "Mesoscale structures at complex fluid–fluid interfaces: A novel lattice Boltzmann/molecular dynamics coupling," Soft Matter **9**, 10092–10107.

Shan, X., and H. Chen, 1993, "Lattice Boltzmann model for simulating flows with multiple phases and components," Phys. Rev. E **47**, 1815.

Shan, X., and H. Chen, 1994, "Simulation of nonideal gases and liquid-gas phase transitions by the lattice Boltzmann equation," Phys. Rev. E **49**, 2941.

Shan, X., X.-F. Yuan, and H. Chen, 2006, "Kinetic theory representation of hydrodynamics: A way beyond the Navier–Stokes equation," J. Fluid Mech. **550**, 413–441.

Shaw, D. E., *et al.*, 2014, "Anton 2: Raising the Bar for Performance and Programmability in a Special-Purpose Molecular Dynamics Supercomputer," in *SC14: International Conference for High Performance Computing, Networking, Storage and Analysis* (ACM, New York), pp. 41–53.

Shaw, D. E., *et al.*, 2008, "Anton, a special-purpose machine for molecular dynamics simulation," Commun. ACM **51**, 91–97.

Shi, X., G. Lin, J. Zou, and D. A. Fedosov, 2013, "A lattice Boltzmann fictitious domain method for modeling red blood cell deformation and multiple-cell hydrodynamic interactions in flow," Int. J. Numer. Methods Fluids **72**, 895–911.

Sofonea, V., A. Lamura, G. Gonnella, and A. Cristea, 2004, "Finite-difference lattice Boltzmann model with flux limiters for liquid-vapor systems," Phys. Rev. E **70**, 046702.

Sterpone, F., *et al.*, 2014, "The OPEP protein model: From single molecules, amyloid formation, crowding and hydrodynamics to DNA/RNA systems," Chem. Soc. Rev. **43**, 4871–4893.

Storm, A. J., C. Storm, J. Chen, H. Zandbergen, J.-F. Joanny, and C. Dekker, 2005, "Fast DNA translocation through a solid-state nanopore," Nano Lett. **5**, 1193–1197.

Struchtrup, H., and M. Torrilhon, 2003, "Regularization of Grad's 13 moment equations: Derivation and linear analysis," Phys. Fluids **15**, 2668–2680.

Struchtrup, H., and M. Torrilhon, 2007, "H theorem, regularization, and boundary conditions for linearized 13 moment equations," Phys. Rev. Lett. **99**, 014502.

Succi, S., 2001, *The lattice Boltzmann equation: For fluid dynamics and beyond* (Oxford University Press, New York).

Succi, S., 2002a, "Lattice Boltzmann equation for relativistic quantum mechanics," Phil. Trans. R. Soc. A **360**, 429–436.

Succi, S., 2002b, "Mesoscopic modeling of slip motion at fluid-solid interfaces with heterogeneous catalysis," Phys. Rev. Lett. **89**, 064502.

Succi, S., 2018, *The lattice Boltzmann equation for complex states of flowing matter* (Oxford University Press, New York).

Succi, S., G. Amati, M. Bernaschi, G. Falcucci, M. Lauricella, and A. Montessori, 2019, "Towards Exascale Lattice Boltzmann computing," Comput. Fluids **181**, 107–115.

Succi, S., and P. V. Coveney, 2018, "Big Data: The End of the Scientific Method?," arXiv:1807.09515.

Succi, S., I. V. Karlin, and H. Chen, 2002, "*Colloquium*: Role of the H theorem in lattice Boltzmann hydrodynamic simulations," Rev. Mod. Phys. **74**, 1203.

Succi, S., N. Moradi, A. Greiner, and S. Melchionna, 2014, "Lattice Boltzmann modeling of water-like fluids," Front. Phys. **2**, 22.

Sui, Y., Y. T. Chew, and H. T. Low, 2007, "A lattice Boltzmann study on the large deformation of red blood cells in shear flow," Int. J. Mod. Phys. C **18**, 993–1011.

Sui, Y., Y. T. Chew, P. Roy, Y. P. Cheng, and H. T. Low, 2008, "Dynamic motion of red blood cells in simple shear flow," Phys. Fluids **20**, 112106.

Sun, C., C. Migliorini, and L. L. Munn, 2003, "Red blood cells initiate leukocyte rolling in postcapillary expansions: A lattice Boltzmann analysis," Biophys. J. **85**, 208–222.

Sun, C., and L. L. Munn, 2006, "Influence of erythrocyte aggregation on leukocyte margination in postcapillary expansions: A lattice Boltzmann analysis," Physica A (Amsterdam) **362**, 191–196.

Swift, M. R., E. Orlandini, W. R. Osborn, and J. M. Yeomans, 1996, "Lattice Boltzmann simulations of liquid-gas and binary fluid systems," Phys. Rev. E **54**, 5041.

Takahashi, K., S. N. V. Arjunan, and M. Tomita, 2005, "Space in systems biology of signaling pathways–towards intracellular molecular crowding in silico," FEBS Lett. **579**, 1783–1788.

Tamagawa, M., H. Kaneda, M. Hiramoto, and S. Nagahama, 2009, "Simulation of thrombus formation in shear flows using lattice Boltzmann method," Artificial Organs **33**, 604–610.

Timr, S., S. Melchionna, P. Derreumaux, and F. Sterpone, 2019, "Multi-Scale Simulations Yield Insight into Protein Diffusion and Stability in Crowded Environments," Biophys. J. **116**, 38a.

Tiribocchi, A., M. Lauricella, S. Melchionna, A. Montessori, and S. Succi, 2019, "Curvature dynamics and long-range effects on fluid-fluid interfaces with colloids," Soft Matter (in press).

Tomczak, T., and R. G. Szafran, 2017, "Sparse geometries handling in lattice-Boltzmann method implementation for graphic processors," arXiv:1703.08015.

Voit, E. O., 2013, "Biochemical systems theory: A review," *ISRN Biomathematics* (Hindawi Publishing Corporation, London, UK), Vol. 2013.

Wang, M., and Q. Kang, 2010, "Modeling electrokinetic flows in microchannels using coupled lattice Boltzmann methods," J. Comput. Phys. **229**, 728–744.

Wöhrwag, M., C. Semprebon, A. M. Moqaddam, I. Karlin, and H. Kusumaatmaja, 2018, "Ternary Free-Energy Entropic Lattice Boltzmann Model with a High Density Ratio," Phys. Rev. Lett. **120**, 234501.

Wu, J., and C. K. Aidun, 2010, "Simulating 3D deformable particle suspensions using lattice Boltzmann method with discrete external boundary force," Int. J. Numer. Methods Fluids **62**, 765–783.

Xiong, W., and J. Zhang, 2012, "Two-dimensional lattice Boltzmann study of red blood cell motion through microvascular bifurcation: Cell deformability and suspending viscosity effects," Biomech. Model. Mechanobiol. **11**, 575–583.

Xu, Y.-Q., F.-B. Tian, and Y.-L. Deng, 2013, "An efficient red blood cell model in the frame of IB-LBM and its application," Int. J. Biomathematics **06**, 1250061.

Yun, B. M., L. P. Dasi, C. K. Aidun, and A. P. Yoganathan, 2014, "Computational modelling of flow through prosthetic heart valves using the entropic lattice-Boltzmann method," J. Fluid Mech. **743**, 170–201.

Zhang, R., X. Shan, and H. Chen, 2006, "Efficient kinetic method for fluid simulation beyond the Navier-Stokes equation," Phys. Rev. E **74**, 046703.

Zhou, H. X., G. Rivas, and A. P. Minton, 2008, "Macromolecular crowding and confinement: Biochemical, biophysical, and potential physiological consequences.," Annu. Rev. Biophys. **37**, 375–397.

Zou, Q., and X. He, 1997, "On pressure and velocity boundary conditions for the lattice Boltzmann BGK model," Phys. Fluids **9**, 1591–1598.